



**Faculty of Engineering & Technology  
Electrical & Computer Engineering Department**

**Computer Design Laboratory Project**

**Smart home automation**

---

*Prepared by: ShahEd Jamhour 1180654*

*Instructor: Dr. Abdallatif Abuissa*

*Assistant: Eng. Heba Qadah*

*Section:5*

*Date: May 2021*

## Abstract:

The aim of this project is to build a home automation system using 2 Arduinos, a breadboard and the following components.

- 1.Ultrasonic Sensor.
- 2.Light Dependent Resistor.
- 3.Passive Infrared sensor.
4. Temperature Sensor.
5. DC Motor.
- 6.Bulb.
- 7.Switch.

## Table of Contents

<b>Abstract:</b> .....	1
<b>Theory &amp; Background:</b> .....	3
<b>Design and Implementation:</b> .....	7
Testing .....	12
Conclusion.....	15
<b>Reverences</b> .....	16
Appendix.....	17

## Theory & Background:

This part contains the components and their usage.

### 1. Ultrasonic sensor

The HC-SR04 ultrasonic sensor uses SONAR to determine the distance of an object just like the bats do. It offers excellent non-contact range detection with high accuracy and stable readings in an easy-to-use package from 2 cm to 400 cm or 1” to 13 feet.

The operation is not affected by sunlight or black material, although acoustically, soft materials like cloth can be difficult to detect. It comes complete with ultrasonic transmitter and receiver module. Figure 1 shows an ultrasonic sensor.



Figure (1)

### 2. Light Dependent Resistor – LDR

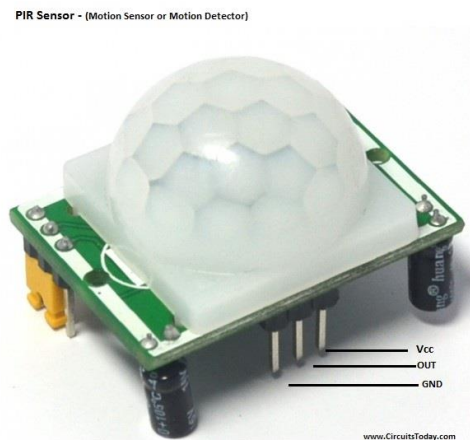
In order to detect the intensity of light or darkness, The LDR is a special type of resistor that allows higher voltages to pass through it (low resistance) whenever there is a high intensity of light, and passes a low voltage (high resistance) whenever it is dark. We can take advantage of this LDR property and use it in this project. Figure 2 shows LDR.



Figure (2)

3. Passive Infrared sensor - PIR

PIR sensor detects a human being moving around within approximately 10m from the sensor. This is an average value, as the actual detection range is between 5m and 12m. PIR are fundamentally made of a pyro electric sensor, which can detect levels of infrared radiation. For numerous essential projects or items that need to discover when an individual has left or entered the area. PIR sensors are incredible, they are flat control and minimal effort, have a wide lens range, and are simple to interface with. Figure 3 shows PIR.



**Figure (3)**

4. A DC motor (Direct Current motor) is the most common type of motor. DC motors normally have just two leads, one positive and one negative. If you connect these two leads directly to a battery, the motor will rotate.

If you switch the leads, the motor will rotate in the opposite direction. Figure 4 shows a DC motor.

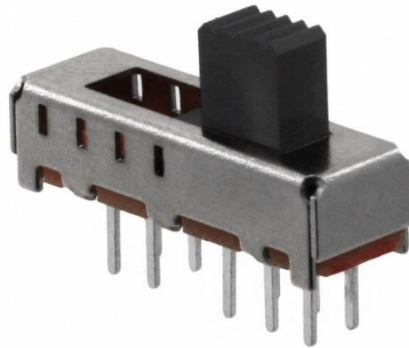


Figure (4)

## 5. Slide Switch

Slide switch is to connect or disconnect the circuit by sliding its switch handle so as to switch the circuit. The common types of slide switch include single pole double throw, single pole triple throw, double pole double throw, and double pole triple throw and so on. Generally, it is used in circuits with a low voltage and features flexibility and stabilization. Slide switches are commonly used in all kinds of instruments/meters equipment, electronic toys and other fields related.

Figure 5 shows a slide switch.



**Slide switch (5)**

## Design and Implementation:

In this project 2 Arduinos were used (As shown in fig 6), the first Arduino will tell the second Arduino, if we are using the manual or automatic mode (By using the switch, if the switch is low , it will work on the automatic mode , if high, it will work on manual mode ).

If in Automatic mode the circuit will start working immediately after starting the simulation, but the manual mode requires the user to use the remote in order to complete any actions.

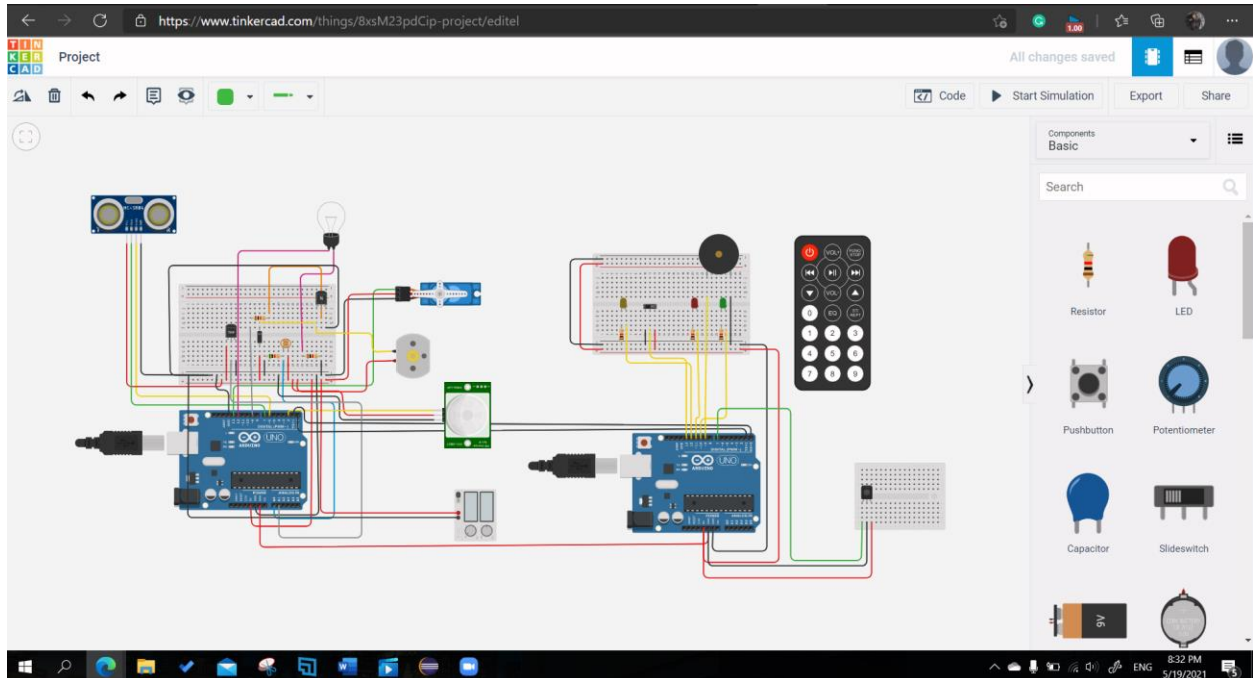


Fig 6

How does the system work?

### 1. Automatic mode

- a. An Ultrasound Sensor was connected in order to measure the distance, if there is something in distance of 50 cm or less, a door (Servo meter) will open for 3 seconds.

```
189 //-----
190 //Get Distance
191 //-----
192 }
193 digitalWrite(trigPin, LOW);
194 // Reads the echoPin, returns the sound wave travel time in microseconds
195 duration = pulseIn(echoPin, HIGH);
196 // Calculating the distance
197 distance = duration * 0.034 / 2; // Speed of sound wave divided by 2 (go and back)
198 // Displays the distance on the Serial Monitor
199
200 //-----
```



```

185
186 //-----
187 // Door Opening part
188 //-----
189 ServoP.write(0) ;
190 if (distance <=50){
191     ServoP.write(120); // door opens
192 }
193 else {
194     //close door
195     ServoP.write(0);
196 }
197 //wait for 3 sec
198 if(millis() >= time_1 + INTERVAL_MESSAGE1){
199     time_1 +=INTERVAL_MESSAGE1;
200 }
201
202
203

```

- b. A Light Dependent Resistor and PIR were connected in order to detect light and movement. If there is no enough light “and” there is a movement then a bulb will be turned on for 1 minute.

```

221 //-----
222 //LDR CHECKS IF THERE ISNT ENOUGH LIGHT
223 //-----
224
225 photocellReading = analogRead(photocellPin);
226 pirStat = digitalRead(pirPin);
227
228
229
230 if (photocellReading < 100 && pirStat == HIGH ) {
231     digitalWrite(BPin, HIGH);
232 }
233 else if (photocellReading < 600 ) {
234     digitalWrite(BPin, LOW);
235 } else {
236     digitalWrite(BPin, LOW);
237 }
238
239
240
241
242

```

- c. Temperature Sensor was connected to measure temperature, if the temperature is more than 30 degree, then a fan (DC Motor) will be turned on, and the fan will go faster as the temperature increases.

```
248 // Check temp
249 }
250
251     sensorInput = analogRead(A1); //read the analog sensor and store it
252     temp = (double)sensorInput / 1024; //find percentage of input reading
253     temp = temp * 5; //multiply by 5V to get voltage
254     temp = temp - 0.5; //Subtract the offset
255     temp = temp * 100; //Convert to degrees
256
257
258     if (temp>=30)
259     {
260         fan = 1; //Trun On Fan.
261         //fan on
262
263         fanSpeed = map(temp, 30, tempMax, 32, 255); // the actual speed of fan
264
265         analogWrite(fan, fanSpeed); // spin the fan at the fanSpeed speed
266     }
267
268     else
269
270     {
271
272         fanSpeed = 0; // fan is not spinning
273
274         digitalWrite(fan, LOW);|
275
```

## 2. Manual mode

- a. As mentioned before if the switch is high it will work in the manual mode.  
LEDS and Buzzers will notify the user if anything needs an action.  
The Arduino#2 will receive different alarms about the situation of Arduino#1, and the user can use the remote control to turn on/off the bulb (By clicking 1), the door (servo motor) (by clicking 2), and the fan (DC motor) (By clicking 3).

```
47 // PROGRAMMING THE REOMTE
48
49 switch (value) {
50     case 2295:
51         Serial.println("1"); // controls Bulb
52         delay(130);
53         break;
54
55     case 34935:
56         delay(130);
57         Serial.println("2"); //controls Servo ( door)
58         break;
59
60     case 18615:
61         delay(100);
62         Serial.println("3"); // controls DC motor aka fan
63         break;
64 }
65
66
67
68
69
70
71
72
73
74
```

Notification will be sent as follows;

If the distance was less than 50 yellow led will be on, when the temperature is valid red led is on , and so one ..

```
78
79  if (Serial.available()) {
80
81      char data  = Serial.read(); // Read a character
82
83      if ( data=='L' ) // turn yellow led if distance is less than 50
84      {
85          digitalWrite(yellowLed, HIGH);
86
87      }
88      else if ( data=='x')
89      {
90          digitalWrite(yellowLed, LOW);
91
92      }
93
94      else if ( data=='y') // turn on bulb when there is enogh light
95      {
96          digitalWrite(greenLed, HIGH);
97
98      }
99      else if ( data=='z') // turn off bulb when there isnt enogh light
100     {
101         digitalWrite(greenLed, LOW); |
102
103     }
104     else if ( data=='T') //Turn red led  on when temp is valid
105     {
106
```

## Testing

### A. Automatic mode

- (Switch is low)
- As shown in fig 7 Distance is more than 50 cm, so the door is moving.
- The light is valid and there is a movement so the light bulb is turned on

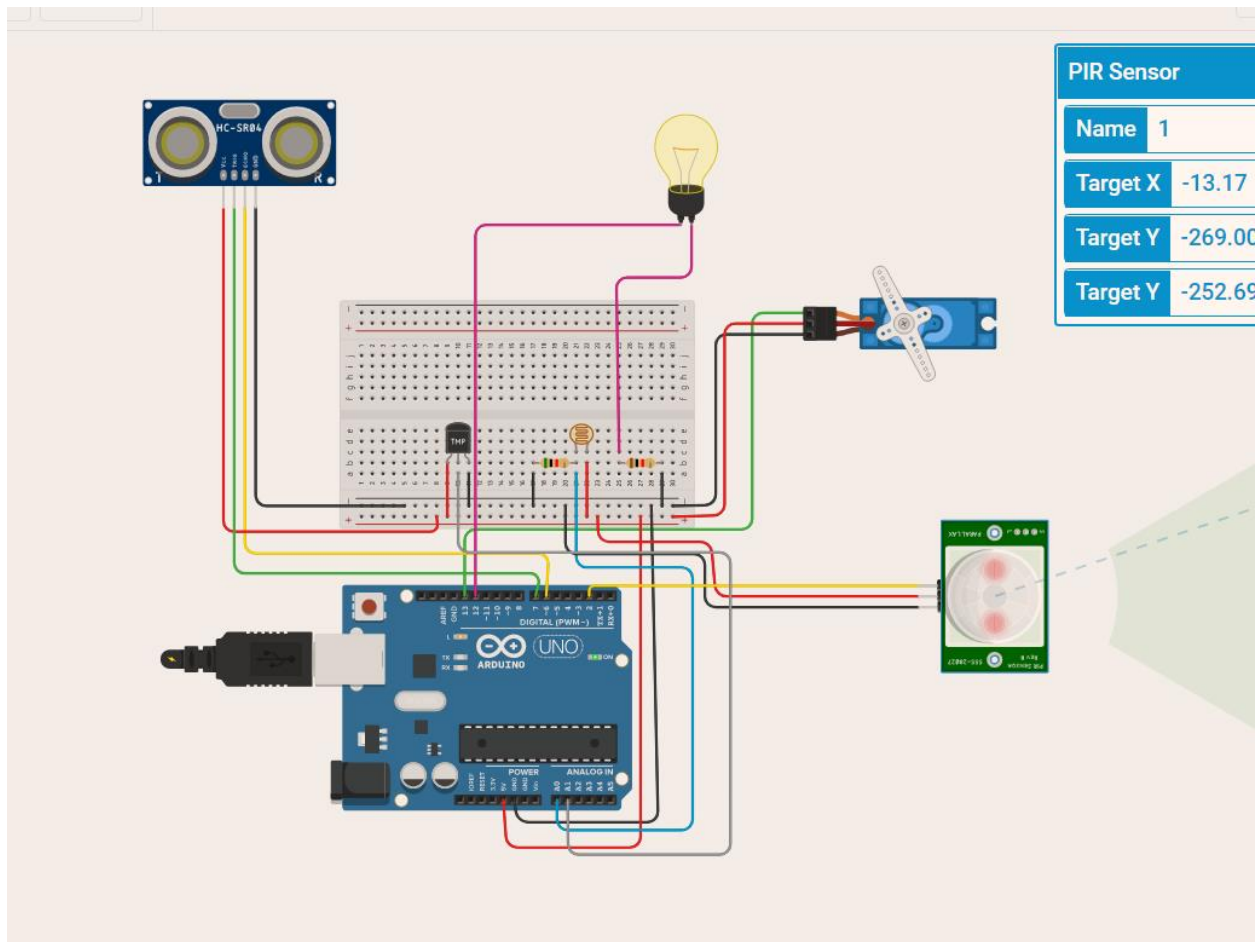


Fig 7

## B.Manual mode

- Switch is high
- If I pressed 1 on the remote it will be printed on the serial as shown in the figure below and the yellow led .
- The bulb is now on since 1 controls the Bulb.

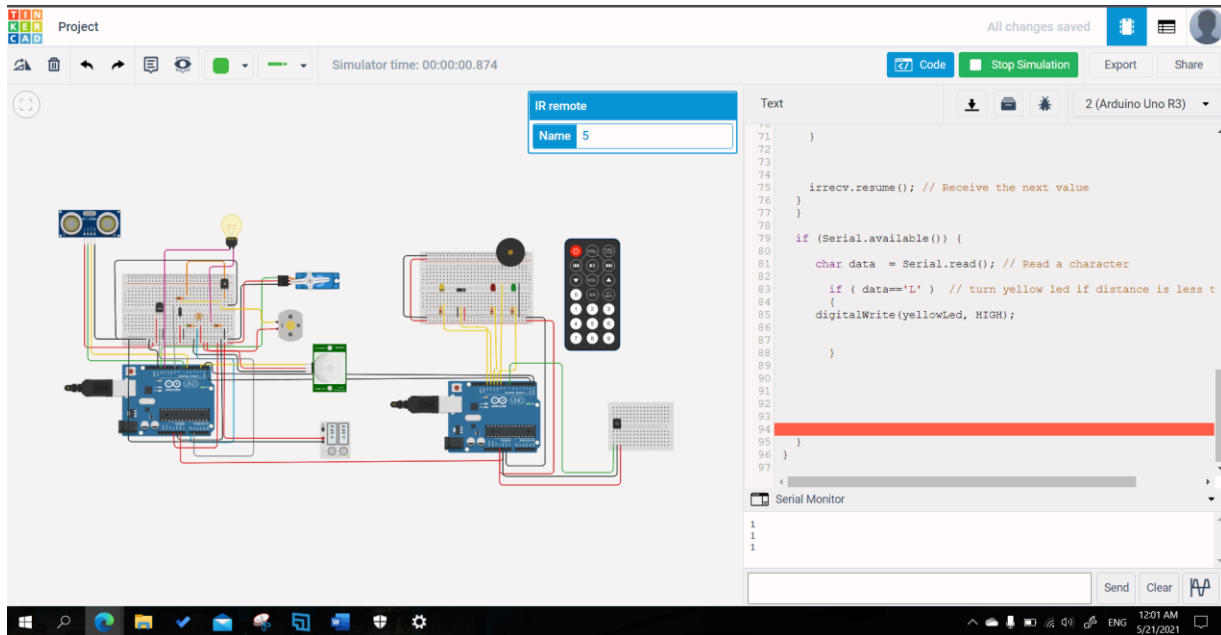


Fig 8

If 2 is pressed the door will start moving

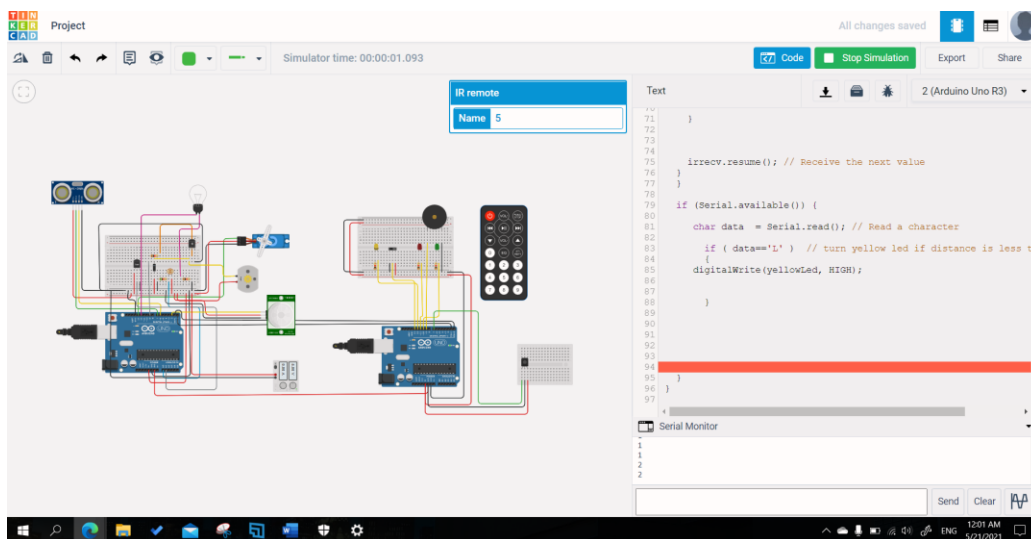
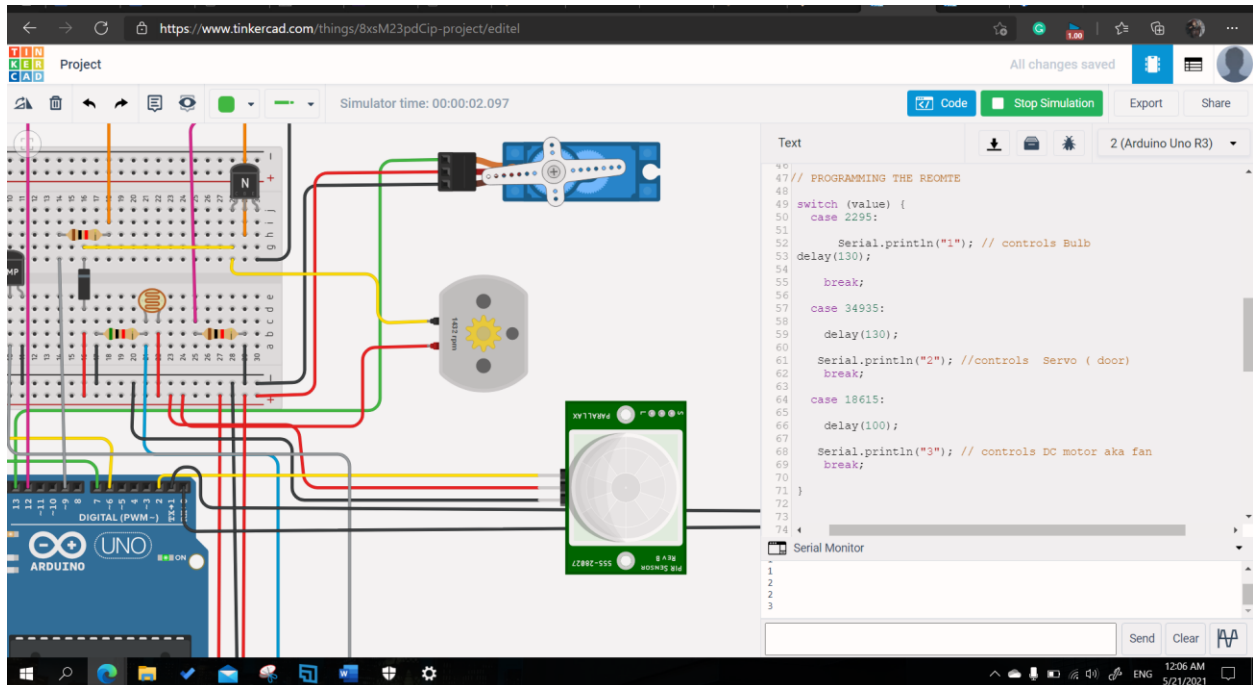


Fig 3

If 3 is pressed then the motor will spin



Link for Full project

[https://www.tinkercad.com/things/8xsM23pdCip-project/editel?sharecode=XH5xfpKZA0vyW1GzZa\\_ZxtV5fUvkMHOBkLQCuyQH-2Y](https://www.tinkercad.com/things/8xsM23pdCip-project/editel?sharecode=XH5xfpKZA0vyW1GzZa_ZxtV5fUvkMHOBkLQCuyQH-2Y)

## Conclusion

In this project I learnt how to build a home automation system using 2 Arduinos and Using the Serial functions.

Also I became more familiar with the following components:

- 1.Ultrasonic Sensor.
- 2.Light Dependent Resistor.
- 3.Passive Infrared sensor.
4. Temperature Sensor.
5. DC Motor.
- 6.Bulb.
- 7.Switch.

.



## Reverences

- [1] [Sweep | Arduino](#) accessed on 19-5-2021
- [2] [Using A TMP36 Temperature Sensor With Arduino - BC Robotics \(bc-robotics.com\)](#) accessed on 19-5-2021
- [3] [Arduino - PIR Sensor - Tutorialspoint](#) accessed on 20-5-2021
- [4] [How to Use an LDR Sensor With Arduino | Arduino | Maker Pro](#) accessed on 20-5-2021
- [5] [Arduino Infrared Remote Tutorial : 7 Steps - Instructables](#) accessed on 20-5-2021

## Appendix

### Full code

#### Arduino #1

```
1 // ----- //
2 #include <Servo.h> //includes the Servo library
3 #define echoPin 6 // attach pin D2 Arduino to pin Echo of HC-SR04
4 #define trigPin 7 //attach pin D3 Arduino to pin Trig of HC-SR04
5 #define INTERVAL_MESSAGE1 3000
6 #define INTERVAL_MESSAGE2 60000
7 //-----+-----
8 //Time Vars
9 //-----
10 unsigned long time_1 = 0;
11 unsigned long time_2 = 0;
12 //-----
13 // Distance variables
14 //-----
15 long duration; // variable for the duration of sound wave travel
16 int distance; // variable for the distance measurement
17
18 //-----
19 // Servo motor
20 Servo ServoP;
21 //-----
22 //PhotoCell and LED photocellPin = 0
23 const int ldrPin = A0;
24 int photocellPin = 0; // the cell and 10K pulldown are connected to a0
25 int photocellReading; // the analog reading from the sensor divider
26 int BPin = 12;
27 //-----
28
29 // PIR
30 int pirPin = 2;
31 int pirStat = 0;
32 //-----
33
34 //Temp Sensor
35 //-----
36 int sensePin = A1;
37 int sensorInput; //The variable we will use to store the sensor input
38 double temp; //The variable we will use to store temperature in degrees.
39 int tempMax = 250;
40 //-----
41 // Motor variables
42 //-----
43
44 int fan = 9; // the pin where fan is connected
45 int fanSpeed = 0;
46
47
48
49 //-----
50 int period = 3000;
51 unsigned long time_now = 0;
52 //-----
53
```

```

54 void setup() {
55     Serial.begin(9600); // // Serial Communication is starting with 9600 of baudrate speed
56     pinMode(trigPin, OUTPUT); // Sets the trigPin as an OUTPUT
57     pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT
58     pinMode(BPin, OUTPUT);
59     pinMode(pirPin, INPUT);
60     ServoP.attach(13);
61     // pinMode(motorPin, OUTPUT);
62     pinMode(fan, OUTPUT);
63
64
65 }
66
67 void loop() {
68     digitalWrite(trigPin, LOW);
69     // Reads the echoPin, returns the sound wave travel time in microseconds
70     duration = pulseIn(echoPin, HIGH);
71     // Calculating the distance
72     distance = duration * 0.034 / 2; // Speed of sound wave divided by 2 (go and back)
73     // Displays the distance on the Serial Monitor
74
75
76     ServoP.write(0) ;
77     if (distance <=50){
78         Serial.println('L'); // Distance Less than 50
79
80
81     }
82
83     else {
84         Serial.println('x'); // invalid distance
85
86
87     //-----
88
89
90     photocellReading = analogRead(photocellPin);
91     pirStat = digitalRead(pirPin);
92
93
94
95     if (photocellReading < 100 && pirStat == HIGH ) {
96         Serial.println('y'); //Bulb on
97
98
99     }
100     else {
101         Serial.println( 'z'); // Bulb off
102
103
104     }
105
106

```

```

109     sensorInput = analogRead(A1); //read the analog sensor and store it
110     temp = (double)sensorInput / 1024; //find percentage of input reading
111     temp = temp * 5; //multiply by 5V to get voltage
112     temp = temp - 0.5; //Subtract the offset
113     temp = temp * 100; //Convert to degrees
114
115
116     if (temp>=30)
117     {
118         Serial.println ('T'); // Valid temp
119     }
120
121     else
122     {
123         Serial.println ('O'); // not valid temp
124     }
125
126
127
128
129
130 //-----
131
132 if (Serial.available() )
133 {
134     |
135     char data = Serial.read(); // Read a character
136
137     if ( data=='A')
138     {
139
140         //-----
141         time_now = millis();
142         //-----
143
144         // Clears the trigPin condition
145         digitalWrite(trigPin, LOW);
146
147         //-----Delay-----
148         if(millis() >= time_1 + INTERVAL_MESSAGE1){
149             time_1 +=INTERVAL_MESSAGE1;
150         }
151
152         //-----
153         // Sets the trigPin HIGH (ACTIVE)
154         digitalWrite(trigPin, HIGH);
155
156
157         //-----Delay For 3 sec -----
158         if(millis() >= time_1 + INTERVAL_MESSAGE1){
159             time_1 +=INTERVAL_MESSAGE1;
160         }
161         //-----
162         //Get Distance
163         //-----
164         digitalWrite(trigPin, LOW);
165         // Reads the echoPin, returns the sound wave travel time in microseconds
166         duration = pulseIn(echoPin, HIGH);
167         // Calculating the distance
168         distance = duration * 0.034 / 2; // Speed of sound wave divided by 2 (go and back)
169         // Displays the distance on the Serial Monitor
170
171         //-----
172         // Door Opening part
173         //-----
174         ServoP.write(0) ;
175         if (distance <=50){
176             ServoP.write(120); // door opens
177
178         }
179         else {
180             //close door
181             ServoP.write(0);
182
183         }
184         //wait for 3 sec
185         if(millis() >= time_1 + INTERVAL_MESSAGE1){
186             time_1 +=INTERVAL_MESSAGE1;
187

```

```

191
192 //-----
193 //LDR CHECKS IF THERE ISNT ENOUGH LIGHT
194 //-----
195
196 photocellReading = analogRead(photocellPin);
197 pirStat = digitalRead(pirPin);
198
199
200
201 if (photocellReading < 100 && pirStat == HIGH ) {
202     digitalWrite(BPin, HIGH);
203
204 }
205 else if (photocellReading < 600 ) {
206     digitalWrite(BPin, LOW);
207
208 } else {
209     digitalWrite(BPin, LOW);
210
211 }
212
213
214 //1 min delay
215 if(millis() >= time_2 + INTERVAL_MESSAGE2){
216     time_2 +=INTERVAL_MESSAGE2;
217
218
219 // Check temp
220     }
221
222     sensorInput = analogRead(A1); //read the analog sensor and store it
223     temp = (double)sensorInput / 1024; //find percentage of input reading
224     temp = temp * 5; //multiply by 5V to get voltage
225     temp = temp - 0.5; //Subtract the offset
226     temp = temp * 100; //Convert to degrees
227
228
229     if (temp>=30)
230     {
231         fan = 1; //Trun On Fan.
232         //fan on
233
234         fanSpeed = map(temp, 30, tempMax, 32, 255); // the actual speed of fan
235
236         analogWrite(fan, fanSpeed); // spin the fan at the fanSpeed speed
237     }
238
239
240     else
241     {
242         fanSpeed = 0; // fan is not spinning
243
244         digitalWrite(fan, LOW);
245
246

```

```

250
251     else if ( data=='1')
252
253     {
254         if (digitalRead (BPin) == HIGH)
255             {
256                 digitalWrite(BPin, LOW);
257             }
258         else
259             {
260                 digitalWrite(BPin, HIGH);
261             }
262     }
263
264     else if (data=='2' )
265
266
267     {
268         if (ServoP.read() == 0 )
269             {
270                 ServoP.write(180);
271             }
272         else
273             {
274                 ServoP.write(0);
275             }
276
277

```

```

282
283     else if ( data=='3')
284     {
285
286
287         if (digitalRead (fan) == HIGH)
288             {
289                 digitalWrite(fan, LOW);
290             }
291         else
292             {
293                 digitalWrite(fan, HIGH);
294             }
295
296
297

```

## Arduino 2

```
Text
1 #include <IRremote.h>
2
3 int redLed = 11;
4 int yellowLed = 12;
5 int greenLed = 10;
6 int Buzzer = 9;
7 int RECV_PIN = 7;
8 int SwitchP= 13;
9
10 //IR Library
11 IRrecv irrecv(RECV_PIN);
12 decode_results results;
13
14 void setup()
15 {
16     //Set Led Pins
17     pinMode(redLed, OUTPUT);
18     pinMode(yellowLed, OUTPUT);
19     pinMode(greenLed, OUTPUT);
20     pinMode(SwitchP, INPUT);
21
22
23     //Enable serial usage and IR signal
24     Serial.begin(9600);
25     irrecv.enableIRIn();
26
27 }
28
```

```
28
29 void loop()
30 {
31     int State = digitalRead(SwitchP);
32     if ( State==LOW) // IF THE SWITCH IS 0 (LOW) IT WILL WORK ON AUTOMATIC MODE
33     {
34         //Automatic mode
35
36         Serial.println("A");
37
38     }
39     else
40     {
41
42         // MANUAL MODE
43
44         if (irrecv.decode(&results)) {
45             unsigned int value = results.value;
46
47             // PROGRAMMING THE REOMTE
48
49             switch (value) {
50                 case 2295:
51
52                     Serial.println("1"); // controls Bulb
53                     delay(130);
54
55                     break;
```

```

56
57     case 34935:
58
59         delay(130);
60
61         Serial.println("2"); //controls Servo ( door)
62         break;
63
64     case 18615:
65
66         delay(100);
67
68         Serial.println("3"); // controls DC motor aka fan
69         break;
70
71     }
72
73
74
75     irrecv.resume(); // Receive the next value
76 }
77 }
78
74
75     irrecv.resume(); // Receive the next value
76 }
77 }
78
79 if (Serial.available()) {
80
81     char data = Serial.read(); // Read a character
82
83     if ( data=='L' ) // turn yellow led if distance is less than 50
84     {
85         digitalWrite(yellowLed, HIGH);
86
87     }
88
89     else if ( data=='x')
90     {
91         digitalWrite(yellowLed, LOW);
92
93     }
94
95     else if ( data=='y') // turn on bulb when there is enogh light
96     {
97         digitalWrite(greenLed, HIGH);
98
99     }
100
101     else if ( data=='y') // turn on bulb when there is enogh light
102     {
103         digitalWrite(greenLed, HIGH);
104
105     }
106
107     else if ( data=='z') // turn off bulb when there isnt enogh light
108     {
109         digitalWrite(greenLed, LOW);
110
111     }
112
113     else if ( data=='T') //Turn red led on when temp is valid
114     {
115         digitalWrite(redLed, HIGH);
116
117     }
118
119     else if ( data=='o') // //Turn red led off when temp isnt valid
120     {
121         digitalWrite(redLed, LOW);
122
123     }
124
125 }

```