



# ISLAMIC UNIVERSITY

Kushtia, Bangladesh



## ARDUINO PROJECTS ON:

IR Remote Controlled Home Automation System using Arduino  
in tinkercad

Prepared for:

**Mohammad Alamgir Hossain**

Assistant Professor

Department of Computer Science and Engineering

Faculty of Engineering and Technology

Islamic University, Kushtia, Bangladesh

Prepared by:

**Abdul Samad Robin**

Roll No: 21214116

13th batch, Group-A, 1st semester

Session: 2021-2022

Date of submission: 25/08/2022

Department of Computer Science and Engineering  
Islamic University, Kushtia, Bangladesh

## Index for IR Remote Controlled Home Automation System using Arduino

Contain	Title	Page Number
01	What is Arduino Uno	3
02	Figure 1.1: An Arduino Uno R3.	3
03	What is Arduino IR remote Control system?	3
04	Figure 1.1: IR remote control sensor	3
05	How does IR remote control works?	4
06	4 Channel Relay Module	4
07	Figure 1.3: 4 Channel Relay Module	4
08	How we will Proceed toward the project?	5
09	Components Required	6
10	Figure 1.4: Components List	6
11	Figure 1.5: IR Remote Controlled Home Automation System using Arduino Circuit Diagram	7
12	IR Remote Controlled Arduino Code	8-10
13	Project Simulation	11
14	Figure 1.6: Project Simulation using tinkercad.	11

## What is Arduino Uno?

**Arduino:** The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc and initially released in 2010. The board is equipped with sets of digital and analog input/output pins that may be interfaced to various expansion boards and other circuits.

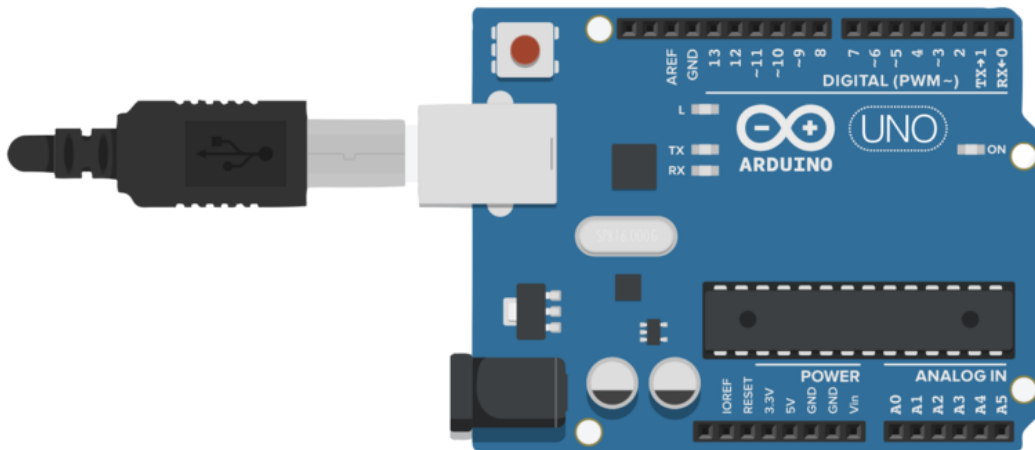


Figure 1.1: An Arduino Uno R3.

## What is Arduino IR remote Control system?

**Arduino IR remote Control system:** in this system, we must have an input remote which is known as the IR transmitter remote which does have many keys and there is also a receiver that receives the data from the remote. Remotely send the different data value on the different button and receiver receive that value. When we press any key the associated led will be glow. And the rest of the LED will not glow. And in the next process, we will add the relay and the ac load with the system so the **Arduino ir remote** can operate the AC load too. The different buttons will associate with the different sources so that we can operate many appliances. That is cool to operate the thing via remote. As you all saw the center lock car key. The car key uses radiofrequency but this system uses IR frequency. We need to make the remote front facing towards the receiver.

### IR transmitter



### IR Receiver



Figure 1.2: IR remote control sensor.

## How does IR remote control works?

**IR remote control works:** IR technology dominant the market of television, home theatre, and many more. In Infrared wireless communication, the light carries some information which transmits by the IR transmitter and receives by the receiver. Which decode that light to some information. There are some buttons given on the remote as we have given below. Each button carries different information. In Arduino IR remote control project. Each information is associated with a different output which is here the led. When we press the key the associated led will start to glow.

**4 Channel Relay Module:** The 4 Channel Relay Module is a convenient board which can be used to control high voltage, high current load such as motor, solenoid valves, lamps and AC load. It is designed to interface with microcontroller such as Arduino, PIC and etc. The relays terminal (COM, NO and NC) is being brought out with screw terminal. It also comes with a LED to indicate the status of relay

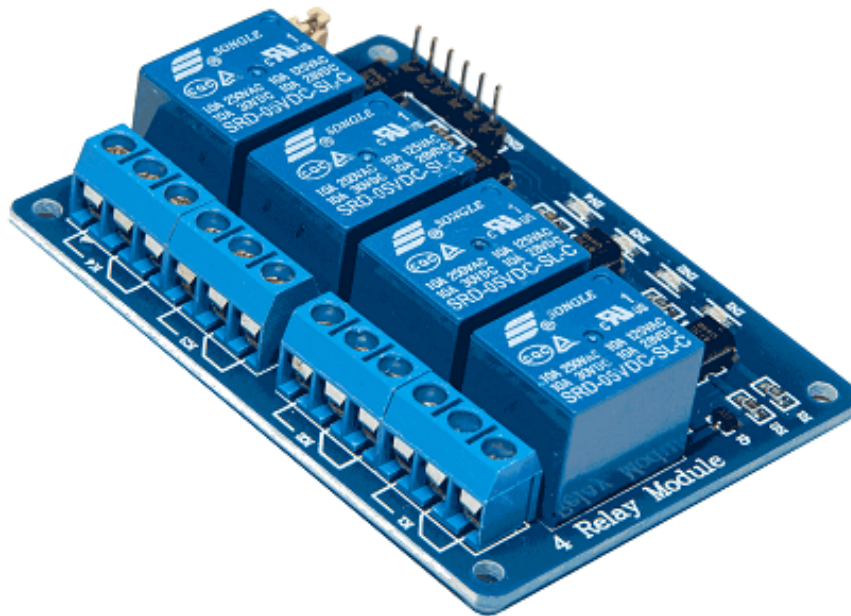


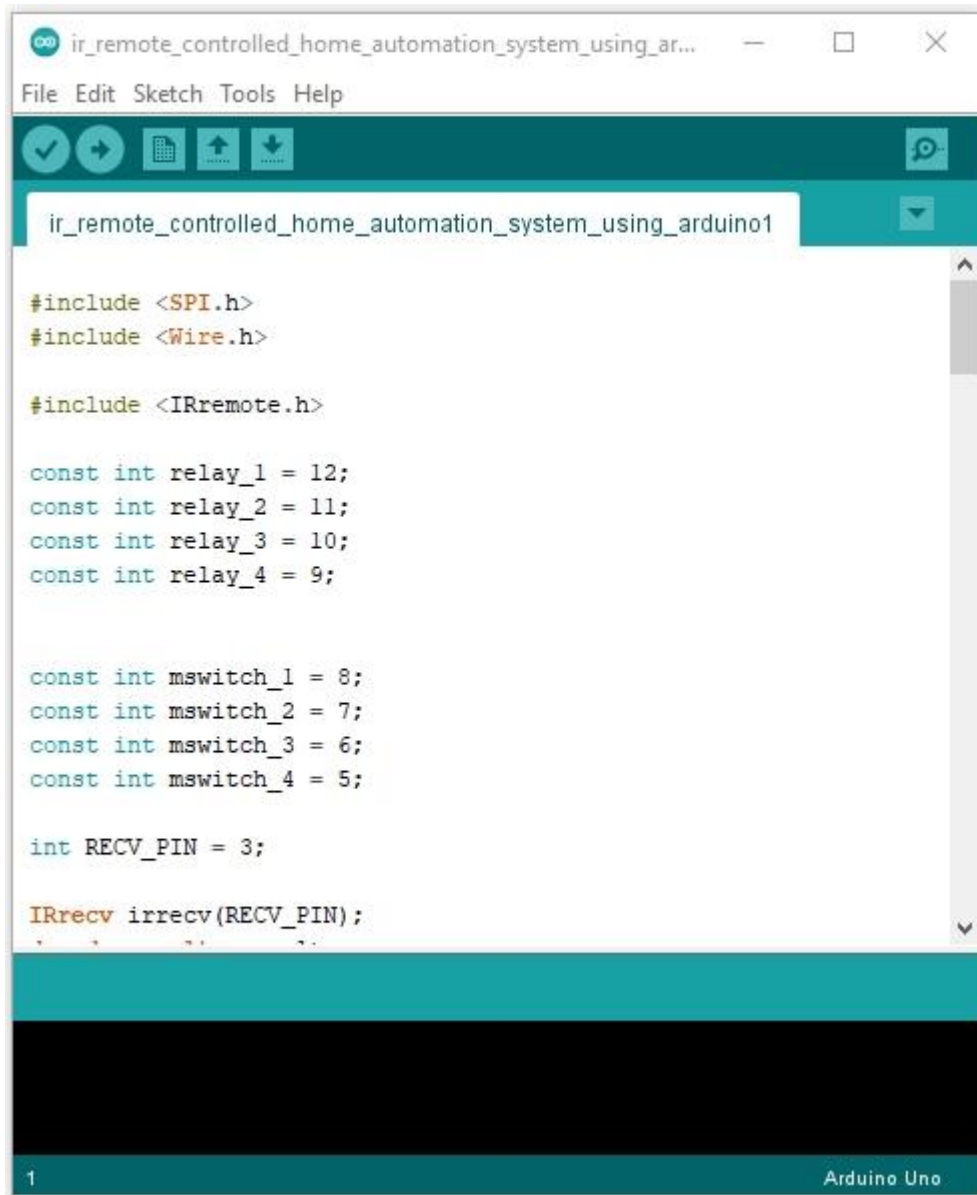
Figure 1.3: 4 Channel Relay Module.

**Specification:**

- Digital output controllable
- Compatible with any 5V microcontroller such as Arduino.
- Rated through-current: 10A (NO) 5A (NC)
- Control signal: TTL level
- Max. switching voltage 250VAC/30VDC
- Max. switching current 10A
- Size: 76mm x 56mm x 17mm

## How I will proceed toward the project?

**Process the project:** First, we will make the basic **IR project** with some Switch's and LEDs which will be controlled by the IR sensor interface with the Arduino, and after that, we will interface the all system with the relay's and the AC sources (I will use the DC source for test purpose) so that we will learn the basic process to make the higher project with the smaller one. If you proceed in the major one direction it will make some errors. That is the way to make the major project we make it step by step. So, first we will make the sensor connection to check the connection and errors and after that, we upload the program into the Arduino and interface the **Arduino with ir sensor**.

A screenshot of the Arduino IDE interface. The title bar shows the file name 'ir\_remote\_controlled\_home\_automation\_system\_using\_ar...'. The menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. The toolbar contains icons for checking, running, and saving. The main text area shows the following code:

```
#include <SPI.h>
#include <Wire.h>

#include <IRremote.h>

const int relay_1 = 12;
const int relay_2 = 11;
const int relay_3 = 10;
const int relay_4 = 9;

const int mswitch_1 = 8;
const int mswitch_2 = 7;
const int mswitch_3 = 6;
const int mswitch_4 = 5;

int RECV_PIN = 3;

IRrecv irrecv(RECV_PIN);
```

The status bar at the bottom indicates '1' and 'Arduino Uno'.

Then how we can check the output at the Arduino. We have to download the Arduino and open the serial monitor of the software and then check the output. Now, according to the output we will use the value. On the **ir remote** we can see there are many of the buttons. When we pressed the button over the remote every time we get a different value according to the value we will assign the output. For example, if we press button A the serial monitor shows "RRR" value then we assign this value in the database if the value is equal to the "RRR" then the red led will be on. Same as this we will assign all led and buttons.

## Components Required:

- 1 Arduino Uno R3
- 1 IR Sensor module
- Jumper wire
- 4 Switch's
- Power Supply
- 4 LED
- Relay module 4 channel
- 4 Resistor
- 1 Light bulb, 1 DC Motor , 1 Hobby Gear motor, 1 Vibration Motor

Name	Quantity	Component
U Arduino	1	Arduino Uno R3
S Switch 1 S Switch 2 S Switch 3 S Switch 4	4	Pushbutton
K Relay 1 K Relay 2 K Relay 3 K Relay 4	4	Relay SPDT
U Sensor	1	IR sensor
P Power Source	1	12 , 5 Power Supply
L Tube Bulb	1	Light bulb
M Motor	1	DC Motor
M Gearmotor	1	Hobby Gearmotor
M Vibration Motor	1	Vibration Motor
D Switch Number 1 LED D Switch Number 3 LED D Switch Number 2 LED D Switch Number 4 LED	4	Green LED
R Resistor 3 for LED 3 R Resistor 2 for LED 2 R Resistor 1 for LED 1 R Resistor 4 for LED 4	4	1 k $\Omega$ Resistor

Figure 1.4: Components List.

# IR Remote Controlled Home Automation System using Arduino Circuit Diagram:

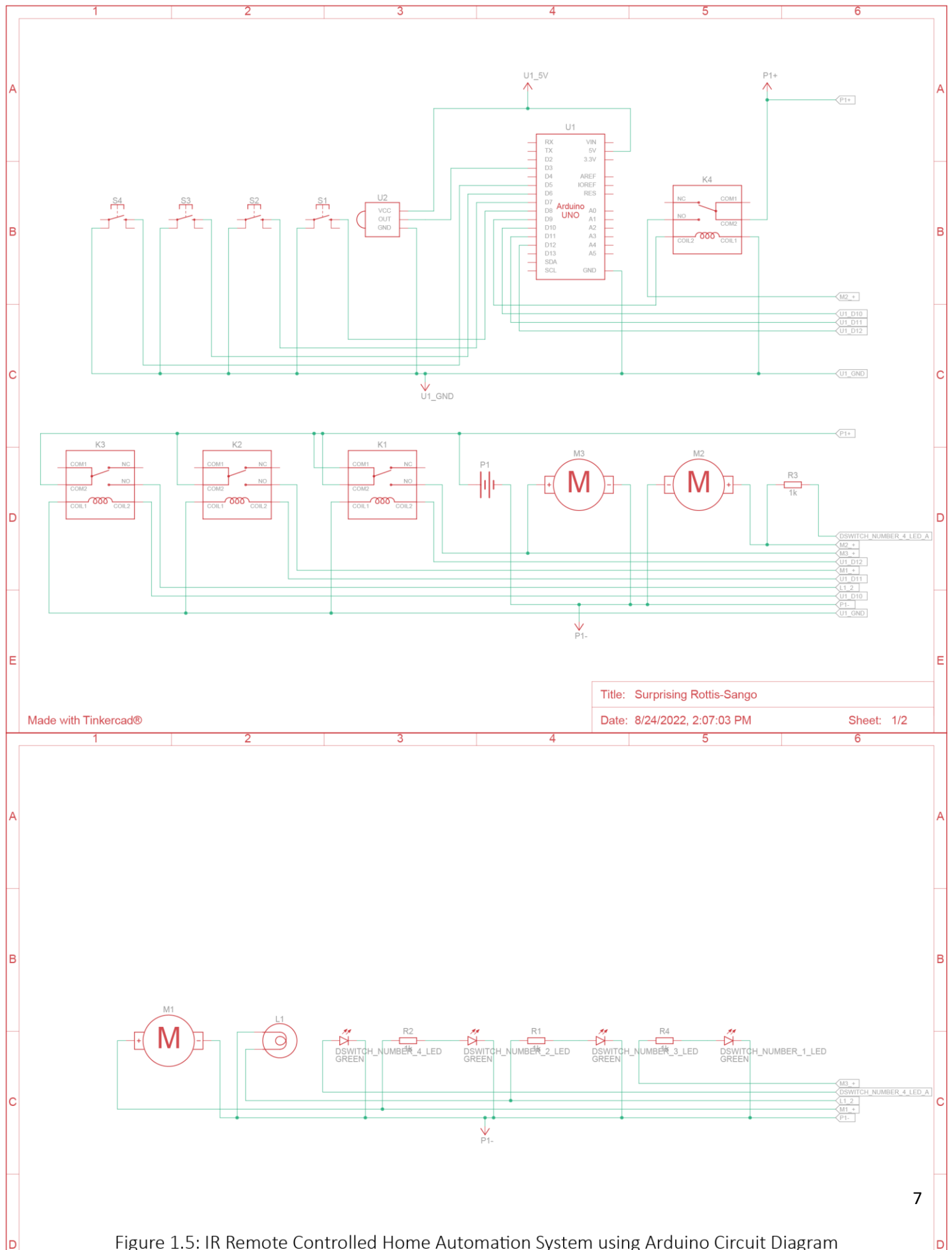


Figure 1.5: IR Remote Controlled Home Automation System using Arduino Circuit Diagram

## IR Remote Controlled Arduino Code:

```
#include <SPI.h>
#include <Wire.h>

#include <IRremote.h>

const int relay_1 = 12;
const int relay_2 = 11;
const int relay_3 = 10;
const int relay_4 = 9;

const int mswitch_1 = 8;
const int mswitch_2 = 7;
const int mswitch_3 = 6;
const int mswitch_4 = 5;

int RECV_PIN = 3;

IRrecv irrecv(RECV_PIN);
decode_results results;

int toggleState_1 = 0;
int toggleState_2 = 0;
int toggleState_3 = 0;
int toggleState_4 = 0;

void setup() {

  Serial.begin(9600);
  irrecv.enableIRIn();

  pinMode(relay_1, OUTPUT);
  pinMode(relay_2, OUTPUT);
  pinMode(relay_3, OUTPUT);
  pinMode(relay_4, OUTPUT);

  pinMode(mswitch_1, INPUT_PULLUP);
  pinMode(mswitch_2, INPUT_PULLUP);
  pinMode(mswitch_3, INPUT_PULLUP);
  pinMode(mswitch_4, INPUT_PULLUP);
}

void relayOnOff(int relay){

  switch(relay){
    case 1:
      if(toggleState_1 == 0){
        digitalWrite(relay_1, HIGH); // turn on relay 1
        toggleState_1 = 1;
      }
    }
```



```

        else{
            digitalWrite(relay_1, LOW); // turn off relay 1
            toggleState_1 = 0;
        }
        delay(100);
break;
case 2:
    if(toggleState_2 == 0){
        digitalWrite(relay_2, HIGH); // turn on relay 2
        toggleState_2 = 1;
    }
    else{
        digitalWrite(relay_2, LOW); // turn off relay 2
        toggleState_2 = 0;
    }
    delay(100);
break;
case 3:
    if(toggleState_3 == 0){
        digitalWrite(relay_3, HIGH); // turn on relay 3
        toggleState_3 = 1;
    }else{
        digitalWrite(relay_3, LOW); // turn off relay 3
        toggleState_3 = 0;
    }
    delay(100);
break;
case 4:
    if(toggleState_4 == 0){
        digitalWrite(relay_4, HIGH); // turn on relay 4
        toggleState_4 = 1;
    }
    else{
        digitalWrite(relay_4, LOW); // turn off relay 4
        toggleState_4 = 0;
    }
    delay(100);
break;

default : break;
}

}

void loop() {

    if (digitalRead(mswitch_1) == LOW){
        delay(200);
        relayOnOff(1);
    }

```

```

else if (digitalRead(mswitch_2) == LOW){
    delay(200);
    relayOnOff(2);
}
else if (digitalRead(mswitch_3) == LOW){
    delay(200);
    relayOnOff(3);
}
else if (digitalRead(mswitch_4) == LOW){
    delay(200);
    relayOnOff(4);
}

if (irrecv.decode(&results)) {
    switch(results.value){
        case 0xFD08F7:
            relayOnOff(1);
            break;
        case 0xFD8877:
            relayOnOff(2);
            break;
        case 0xFD48B7:
            relayOnOff(3);
            break;
        case 0xFD28D7:
            relayOnOff(4);
            break;
        default : break;
    }
    irrecv.resume();
}
}

```

## Project Simulation using tinkercad:

IR Remote Controlled Home Automation System using Arduino

React ❤️ 0 ⚙️ ✕

design by:  
**Abdul Samad Robin**

Edited 8/24/22, Created 8/21/22

Tinker this

Simulate Add Image

Copy link Design is visible only to you. Change visibility to share

This is an original of IR Remote Controlled Home Automation System using Arduino by Abdul Samad Robin.

<https://www.tinkercad.com/things/2IVmVEnXMWw>

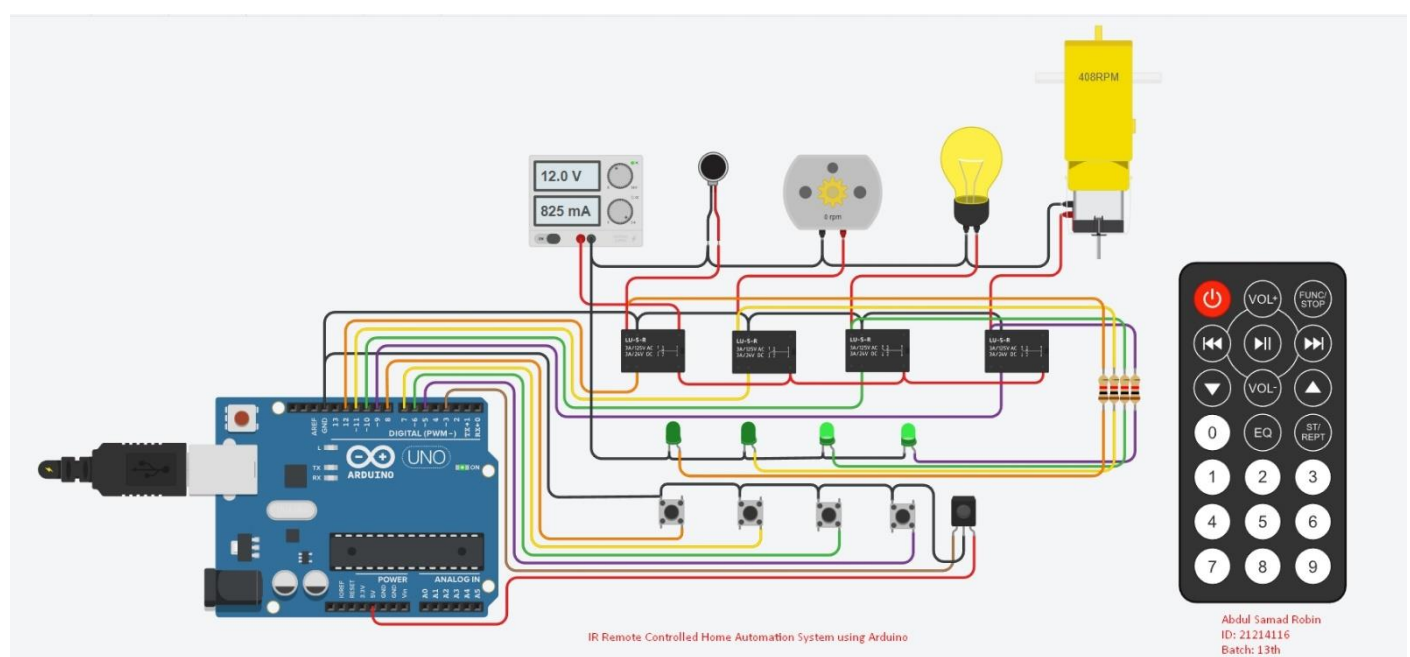


Figure 1.6: Project Simulation using tinkercad.