1. **To Examine the Internet of Things possible applications.**

**10 Real World Applications of IoT**

**1. Smart Home**

With IoT creating the buzz, ‘Smart Home’ is the most searched IoT associated feature on Google. But, what is a Smart Home?

Wouldn’t you love if you could switch on air conditioning before reaching home or switch off lights even after you have left home? Or unlock the doors to friends for temporary access even when you are not at home. Don’t be surprised with IoT taking shape companies are building products to make your life simpler and convenient.

Smart Home has become the revolutionary ladder of success in the residential spaces and it is predicted Smart homes will become as common as smartphones.

The cost of owning a house is the biggest expense in a homeowner’s life. Smart Home products are promised to save time, energy and money. With Smart home companies like Nest, Ecobee, Ring and August, to name a few, will become household brands and are planning to deliver a never seen before experience

**2. Wearables**

Wearables have experienced a explosive demand in markets all over the world. Companies like Google, Samsung have invested heavily in building such devices. But, how do they work?

Wearable devices are installed with sensors and softwares which collect data and information about the users. This data is later pre-processed to extract essential insights about user.

These devices broadly cover fitness, health and entertainment requirements. The pre-requisite from internet of things technology for wearable applications is to be highly energy efficient or ultra-low power and small sized.

**3. Connected Cars**

The automotive digital technology has focused on optimizing vehicles internal functions. But now, this attention is growing towards enhancing the in-car experience.

A connected car is a vehicle which is able to optimize it’s own operation, maintenance as well as comfort of passengers using onboard sensors and internet connectivity.

Most large auto makers as well as some brave startups are working on connected car solutions. Major brands like Tesla, BMW, Apple, Google are working on bringing the next revolution in automobiles.

**4. Industrial Internet**

Industrial Internet is the new buzz in the industrial sector, also termed as Industrial Internet of Things ( IoT ). It is empowering industrial engineering with sensors, software and big data analytics to create brilliant machines.

According to Jeff Immelt, CEO, GE Electric, IoT is a “beautiful, desirable and investable” asset. The driving philosophy behind IoT is that, smart machines are more accurate and consistent than humans in communicating through data. And, this data can help companies pick inefficiencies and problems sooner.

IoT holds great potential for quality control and sustainability. Applications for tracking goods, real time information exchange about inventory among suppliers and retailers and automated delivery will increase the supply chain efficiency. According to GE the improvement industry productivity will generate $10 trillion to $15 trillion in GDP worldwide over next 15 years.

**5. Smart Cities**

Smart city is another powerful application of IoT generating curiosity among world’s population. Smart surveillance, automated transportation, smarter energy management systems, water distribution, urban security and environmental monitoring all are examples of internet of things applications for smart cities.

IoT will solve major problems faced by the people living in cities like pollution, traffic congestion and shortage of energy supplies etc. Products like cellular communication enabled Smart Belly trash will send alerts to municipal services when a bin needs to be emptied.

By installing sensors and using web applications, citizens can find free available parking slots across the city. Also, the sensors can detect meter tampering issues, general malfunctions and any installation issues in the electricity system.

**6. IoT in agriculture**

With the continous increase in world’s population, demand for food supply is extremely raised. Governments are helping farmers to use advanced techniques and research to increase food production. Smart farming is one of the fastest growing field in IoT.

Farmers are using meaningful insights from the data to yield better return on investment. Sensing for soil moisture and nutrients, controlling water usage for plant growth and determining custom fertilizer are some simple uses of IoT.

**7. Smart Retail**

The potential of IoT in the retail sector is enormous. IoT provides an opportunity to retailers to connect with the customers to enhance the in-store experience.

Smart phones will be the way for retailers to remain connected with their consumers even out of store. Interacting through Smartphones and using Beacon technology can help retailers serve their consumers better. They can also track consumers path through a store and improve store layout and place premium products in high traffic areas.

**8. Energy Engagement**

Power grids of the future will not only be smart enough but also highly reliable. Smart grid concept is becoming very popular all over world.

The basic idea behind the smart grids is to collect data in an automated fashion and analyze the behavior or electricity consumers and suppliers for improving efficiency as well as economics of electricity use.

Smart Grids will also be able to detect sources of power outages more quickly and at individual household levels like near by solar panel, making possible distributed energy system.

**9. IOT in Healthcare**

Connected healthcare yet remains the sleeping giant of the Internet of Things applications. The concept of connected healthcare system and smart medical devices bears enormous potential not just for companies, but also for the well-being of people in general.

Research shows IoT in healthcare will be massive in coming years. IoT in healthcare is aimed at empowering people to live healthier life by wearing connected devices.

The collected data will help in personalized analysis of an individual’s health and provide tailor made strategies to combat illness.

**10. IoT in Poultry and Farming**

Livestock monitoring is about animal husbandry and cost saving. Using IoT applications to gather data about the health and well being of the cattle, ranchers knowing early about the sick animal can pull out and help prevent large number of sick cattle. With the help of the collected data and ranchers can increase the poultry production.

1. **Switch ON/OFF LED Using ARDUINO UNO**.

**Required Components:**

1. Arduino Uno R3 + Arduino data cable I already have this broad
2. USB 2.0 Male to B Male cable
3. Arduino Uno R3 Software I have also downloaded
4. One LED light “5MM”
5. 300 Ohm resister
6. Jumper wire only two (Male to Male)
7. Bread broad
8. Visual Studio Any versions

**Source Code For Ardunio:**

int data;

void setup() {

Serial.begin(9600);

pinMode(13,OUTPUT);

}

void loop() {

if (Serial.available()) {

data = Serial.read();

if (data == 'A') {

digitalWrite(13, HIGH);

} else {

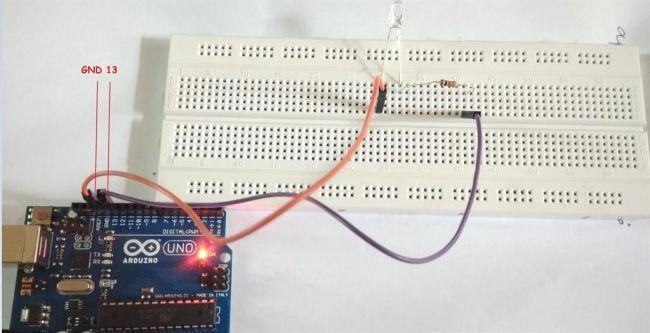
digitalWrite(13, LOW);

}

}

}

**Circuit Diagram:**

****

**Source Code in C# For Visual Studio:**

public Form1() {

InitializeComponent();

serialPort1.Open();

}

private void button1\_Click(object sender, EventArgs e) {

serialPort1.Write("A");

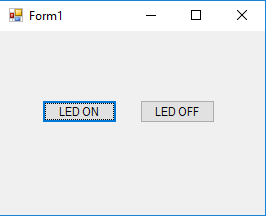
}

private void button2\_Click(object sender, EventArgs e) {

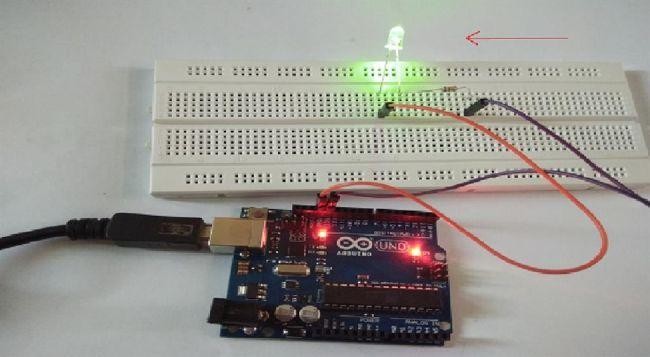
serialPort1.Write("B");

}

**Form Output:**



Click LED ON and OFF; the process is working.



1. **Multiple LED blinks with Arduino.**

**Required Components:**

1. ARDUINO
2. 7 to 8 Jumper Male Pins
3. Bread Board
4. 2 or More LEDs
5. USB Cable – Used to connect ARDUINO with Computer to send commands.

**Source Code For Ardunio:**

#define BaudRate 9600

char inchar; //Will hold the incoming character from the Serial Port.

// Start: I am going to control the below LEDs

int isFanOn = 0;

int isLightOn = 0;

int isBoilerOn = 0;

// End

// Start: Below LED Planes int led1 = 4;

int led2 = 5; int led3 = 6;

// End

void setup()

{

// serial communication Serial.begin(BaudRate);

// prepare the digital output pins pinMode(led1, OUTPUT); pinMode(led2, OUTPUT); pinMode(led3, OUTPUT);

// initially all are off digitalWrite(led1, LOW); digitalWrite(led2, LOW); digitalWrite(led3, LOW);

}

void loop()

{

inchar= Serial.read();

// Serial.println(inchar);

if(inchar==’O’)

{

isFanOn = 0;

isLightOn = 0;

isBoilerOn = 0;

digitalWrite(led1, LOW); digitalWrite(led2, LOW); digitalWrite(led3, LOW);

Serial.println(“O”);

}

if(inchar==’F’ && isFanOn==0)

{

isFanOn = 1; digitalWrite(led1, HIGH); Serial.println(“F1”);

}

else if (inchar==’F’ && isFanOn==1)

{

isFanOn = 0; digitalWrite(led1, LOW); Serial.println(“F0”);

}

if(inchar==’L’ && isLightOn==0)

{

isLightOn = 1; digitalWrite(led2, HIGH); Serial.println(“L1”);

}

else if (inchar==’L’ && isLightOn==1)

{

isLightOn = 0; digitalWrite(led2, LOW); Serial.println(“L0”);

}

if(inchar==’B’ && isBoilerOn==0)

{

isBoilerOn = 1; digitalWrite(led3,HIGH);

Serial.println(“B1”);

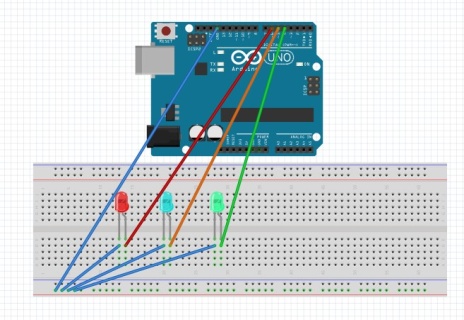
}

else if (inchar==’B’ && isBoilerOn==1)

{

isBoilerOn = 0; digitalWrite(led3, LOW);

Serial.println(“B0”);

**Circuit Diagram:**

**C# Source Code**:

#define BaudRate 9600

char inchar; //Will hold the incoming character from the Serial Port.

// Start: I am going to control the below LEDs

int isFanOn = 0; int isLightOn = 0; int isBoilerOn = 0;

// End

// Start: Below LED Planes int led1 = 4;

int led2 = 5; int led3 = 6;

// End

void setup()

{

// serial communication Serial.begin(BaudRate);

// prepare the digital output pins pinMode(led1, OUTPUT); pinMode(led2, OUTPUT); pinMode(led3, OUTPUT);

// initially all are off digitalWrite(led1, LOW); digitalWrite(led2, LOW); digitalWrite(led3, LOW);

}

void loop()

{

inchar= Serial.read();

// Serial.println(inchar);

if(inchar==’O’)

{

isFanOn = 0;

isLightOn = 0;

isBoilerOn = 0;

digitalWrite(led1, LOW); digitalWrite(led2, LOW); digitalWrite(led3, LOW);

Serial.println(“O”);

}

if(inchar==’F’ && isFanOn==0)

{

isFanOn = 1; digitalWrite(led1, HIGH); Serial.println(“F1”);

}

else if (inchar==’F’ && isFanOn==1)

{

isFanOn = 0; digitalWrite(led1, LOW); Serial.println(“F0”);

}

if(inchar==’L’ && isLightOn==0)

{

isLightOn = 1; digitalWrite(led2, HIGH); Serial.println(“L1”);

}

else if (inchar==’L’ && isLightOn==1)

{

isLightOn = 0; digitalWrite(led2, LOW); Serial.println(“L0”);

}

if(inchar==’B’ && isBoilerOn==0)

{

isBoilerOn = 1; digitalWrite(led3, HIGH);

Serial.println(“B1”);

}

else if (inchar==’B’ && isBoilerOn==1)

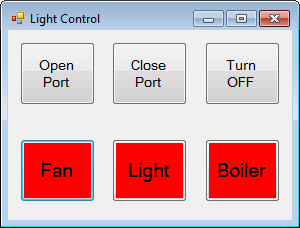
{

isBoilerOn = 0; digitalWrite(led3, LOW); Serial.println(“B0”);

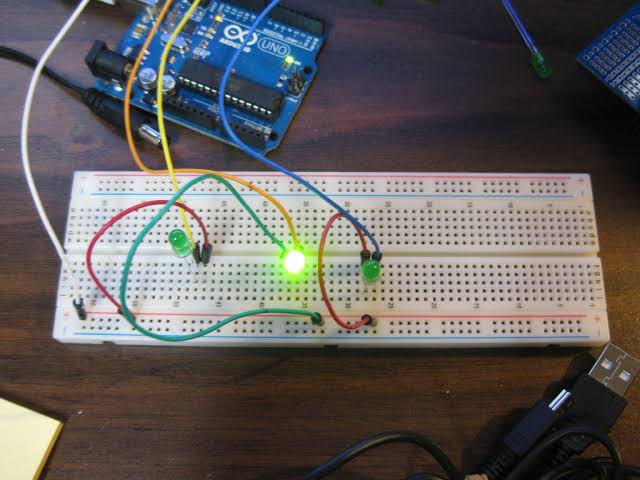
}

}

**Form Output:**



The program is working:

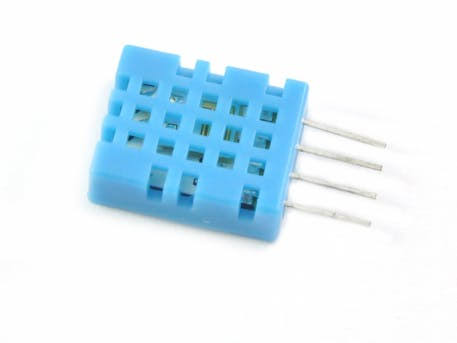


**4. To measure temperature and humidity by DHT11 with Arduino.**

**Required Components:**

1. ARDUINO UNO
2. Jumper Male Pins
3. Bread Board
4. DHT11 Sensor
5. USB Cable – Used to connect ARDUINO with Computer to send commands.

**DHT11 Sensor:**



**Program:**

#include <dht.h>

#define dataPin 8

dht DHT;

void setup()

{

Serial.begin(9600);

}

void loop()

{

int readData = DHT.read11(dataPin);

float t = DHT.temperature;

float h = DHT.humidity;

Serial.print("Temperature = ");

Serial.print(t);

Serial.print(" ");

Serial.print((char)176);

Serial.print("C | ");

Serial.print((t \* 9.0) / 5.0 + 32.0);

Serial.print(" ");

Serial.print((char)176);

Serial.println("F ");

Serial.print("Humidity = ");

Serial.print(h);

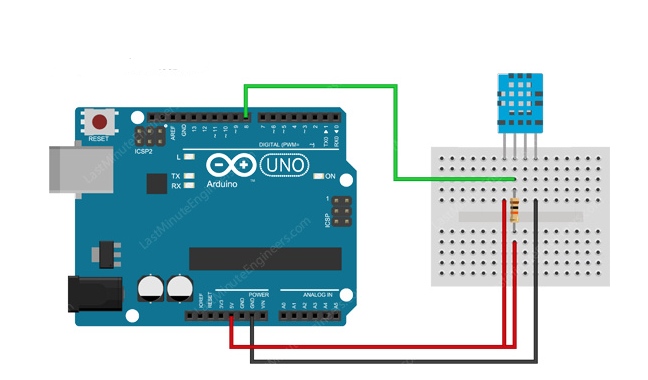
Serial.println(" % ");

Serial.println("");

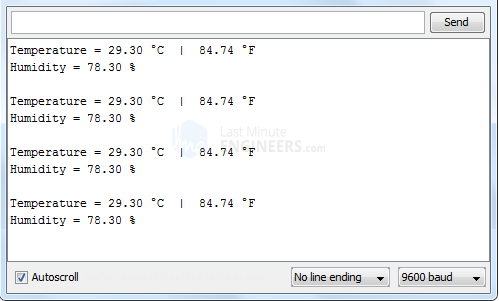
delay(2000);

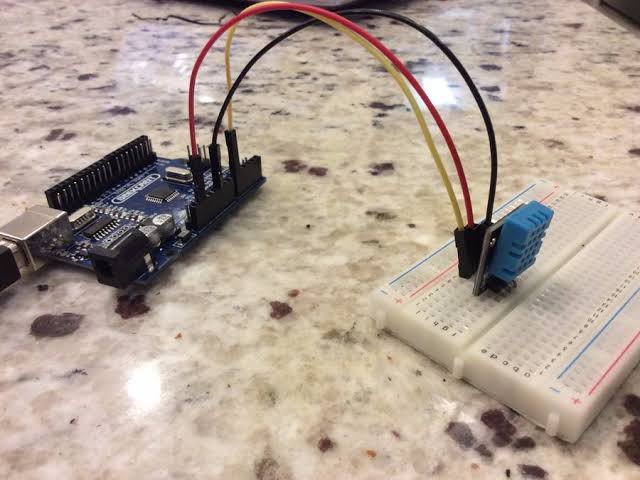
}

**Circuit Diagram:**

****

**Output:**

****



**5. To measure distance between two objects by ultrasonic sensor with Arduino.**

**Required Components:**

1. ARDUINO UNO
2. Jumper Male Pins
3. Bread Board
4. Ultrasonic Sensor
5. USB Cable – Used to connect ARDUINO with Computer to send commands.

**Ultrasonic Sensor:**



**Program:**

int trigPin = 11;

int echoPin = 13;

int led = 11;

void setup() {

Serial.begin(9600);

pinMode(led, OUTPUT);

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

// put your setup code here, to run once:

}

void loop() {

long duration, distance;

digitalWrite(trigPin,HIGH);

delayMicroseconds(1000);

digitalWrite(trigPin, LOW);

duration=pulseIn(echoPin, HIGH);

distance =(duration\*0.034/2);

Serial.print(distance);

Serial.println("CM");

delay(10);

if(distance<=20)

{

digitalWrite(led, HIGH);

Serial.println("Detect Something");

}

else if(distance>10)

Serial.println("Detect Nothing");

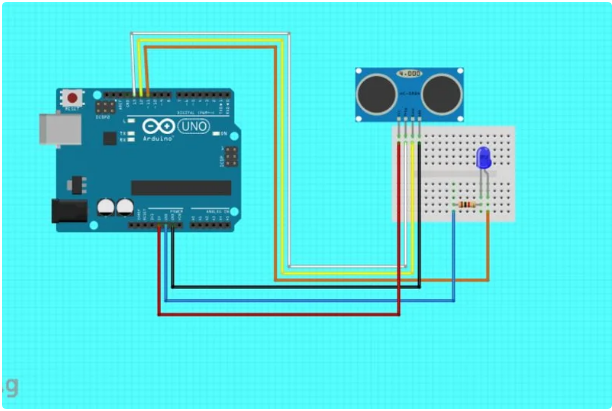
{

digitalWrite(led, LOW);

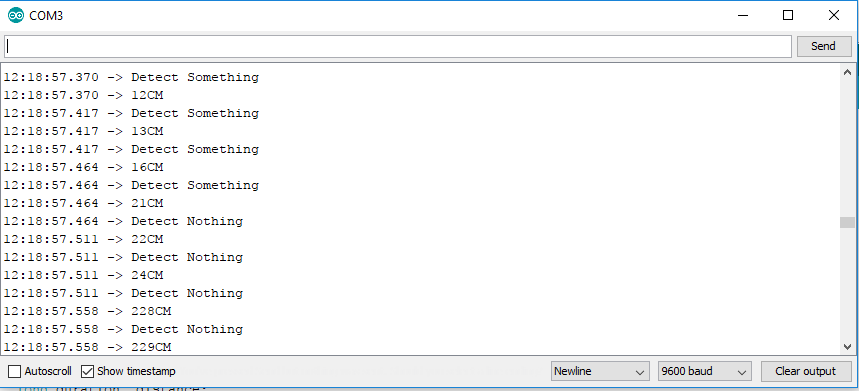
}

}

**Circuit Diagram:**



**Output:**



1. **To detect motion by PIR motion sensor with Arduino.**

**Required Components:**

1. ARDUINO UNO
2. Jumper Wires
3. Bread Board
4. PIR Sensor
5. USB Cable – Used to connect ARDUINO with Computer to send commands.

**PIR Sensor:**

**Program:**

int ledPin = 13;

int inputPin = 2;

int pirState = LOW;

int val = 0;

void setup() {

pinMode(ledPin, OUTPUT);

pinMode(inputPin, INPUT);

Serial.begin(9600);

}

void loop(){

val = digitalRead(inputPin);

if (val == HIGH) {

digitalWrite(ledPin, HIGH);

if (pirState == LOW) {

Serial.println("Motion detected!");

pirState = HIGH;

}

} else {

digitalWrite(ledPin, LOW);

if (pirState == HIGH){

Serial.println("Motion ended!");

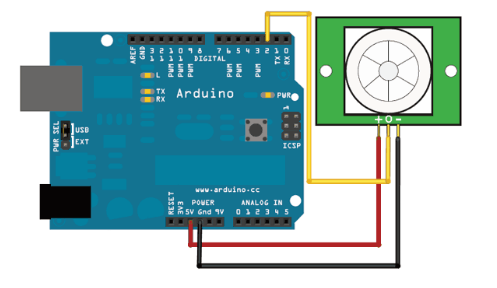
pirState = LOW;

}

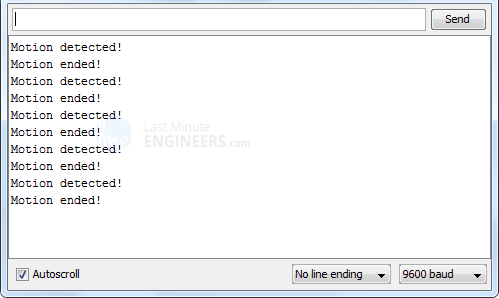
}

}

**Circuit Diagram:**

****

**Output:**

****

1. **To interface RC522 RFID Card Reader with Arduino for restricted environment.**

**Required Components:**

1. ARDUINO UNO
2. Jumper Wires
3. Bread Board
4. RC522 RFID Card Reader
5. USB Cable – Used to connect ARDUINO with Computer to send commands.

**RC522 RFID Card Reader:**



**Program:**

#include <SPI.h>

#include <MFRC522.h>

#define SS\_PIN 10

#define RST\_PIN 9

MFRC522 mfrc522(SS\_PIN, RST\_PIN);

void setup()

{

Serial.begin(9600);

SPI.begin();

mfrc522.PCD\_Init();

Serial.println("Approximate your card to the reader...");

Serial.println();

}

void loop()

{

if ( ! mfrc522.PICC\_IsNewCardPresent())

{

return;

}

if ( ! mfrc522.PICC\_ReadCardSerial())

{

return;

}

Serial.print("UID tag :");

String content= "";

byte letter;

for (byte i = 0; i < mfrc522.uid.size; i++)

{

Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");

Serial.print(mfrc522.uid.uidByte[i], HEX);

content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));

content.concat(String(mfrc522.uid.uidByte[i], HEX));

}

Serial.println();

Serial.print("Message : ");

content.toUpperCase();

if (content.substring(1) == "BD 31 15 2B")

{

Serial.println("Authorized access");

Serial.println();

delay(3000);

else {

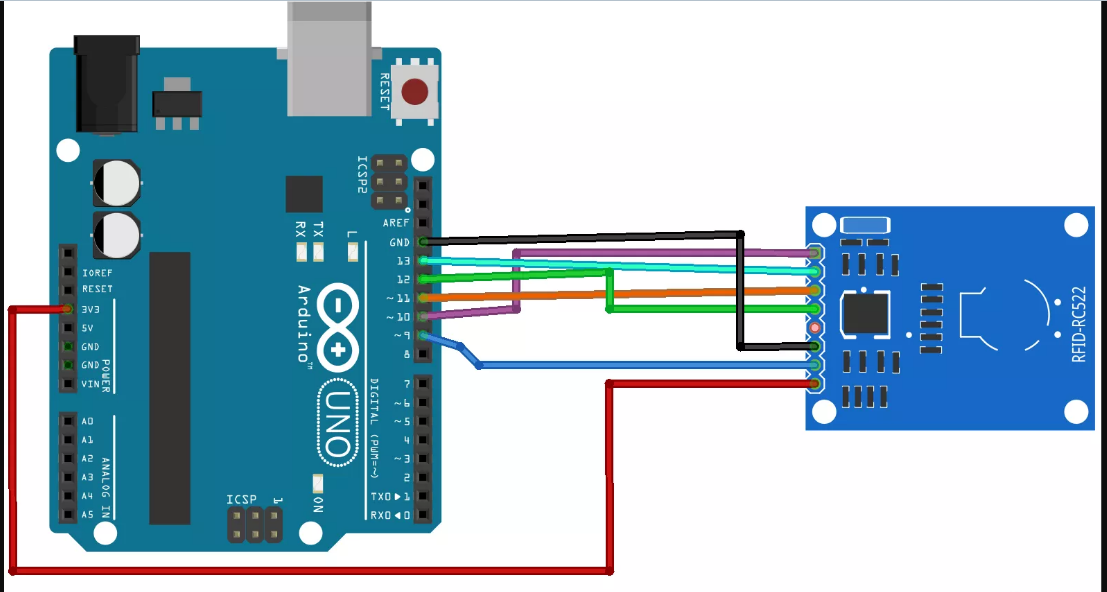
Serial.println(" Access denied");

delay(3000);

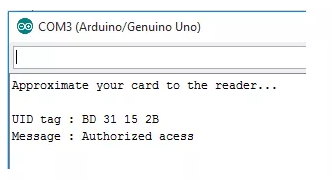
}

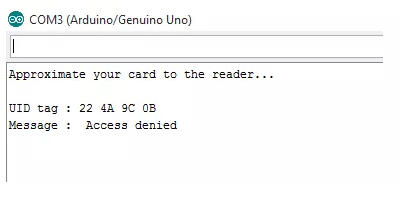
}

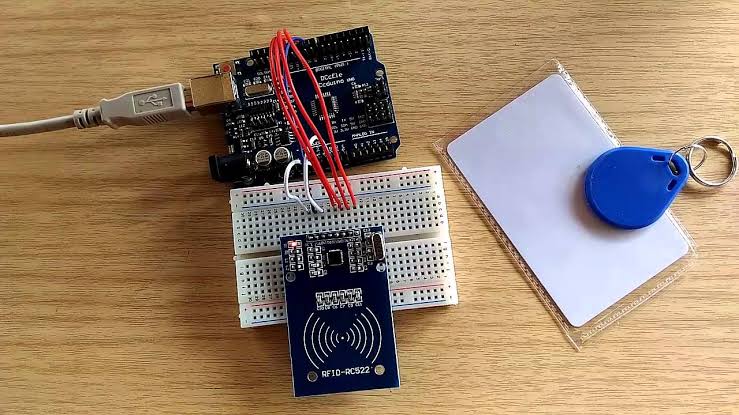
**Circuit Diagram:**

****

**Output:**

****

****

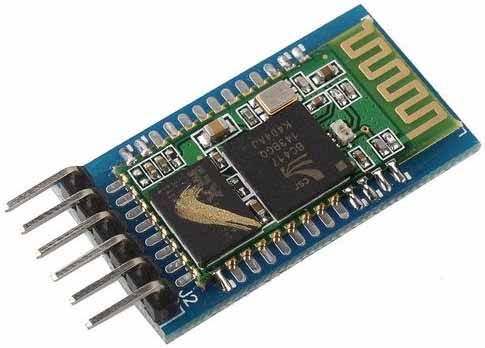


1. **To interface HC05 Bluetooth module by Arduino.**

**Required Components:**

1. HC-05 Bluetooth module (HC-06 will work too)
2. Arduino UNO (Other models will work too)
3. 3 X LED’s
4. 3 X 220 ohm resistors
5. 1k and 2k resistor
6. Connecting wires

**HC05 Bluetooth module:**



**Program:**

int first\_LED = 8;

int second\_LED = 9;

int third\_LED = 10;

int state;

int flag=0; //makes sure that the serial only prints once the state

void setup() {

// sets the pins as outputs:

pinMode(third\_LED, OUTPUT);

pinMode(second\_LED, OUTPUT);

pinMode(first\_LED, OUTPUT);

Serial.begin(9600);

}

void loop() {

//if some date is sent, reads it and saves in state

if(Serial.available() > 0){

state = Serial.read();

flag=0;

}

if (state == '1') {

digitalWrite(first\_LED, HIGH);

if(flag == 0){

Serial.println("First LED ON");

flag=1;

}

}

else if (state == '2') {

digitalWrite(second\_LED, HIGH);

if(flag == 0){

Serial.println("Second LED ON");

flag=1;

}

}

else if (state == '3') {

digitalWrite(third\_LED, HIGH);

if(flag == 0){

Serial.println("Third LED ON");

flag=1;

}

}

else if (state == '0') {

digitalWrite(third\_LED, LOW);

digitalWrite(second\_LED, LOW);

digitalWrite(first\_LED, LOW);

if(flag == 0){

Serial.println("LED: off");

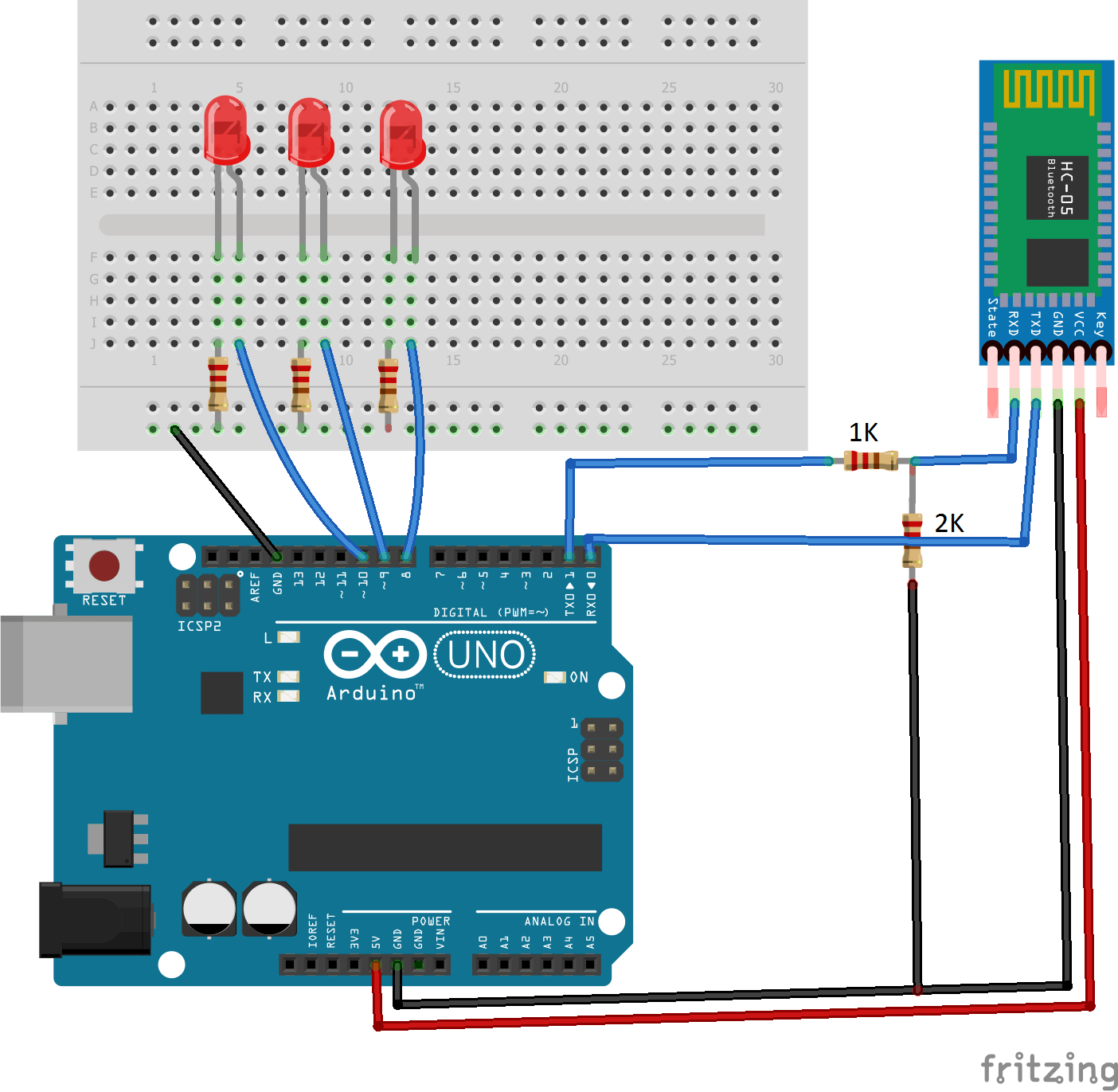
flag=1;

}

}

}

**Circuit Diagram:**



**Output:**



**9. Home automation by Relay with Arduino.**

**Required Components:**

1. ARDUINO UNO
2. Jumper Wires
3. Bread Board
4. HC05 Bluetooth module
5. USB Cable – Used to connect ARDUINO with Computer to send commands.

**ARDUINO Source Code**:

char incomingdata;

void setup() {

pinMode(2, OUTPUT);

pinMode(4, OUTPUT);

pinMode(12, OUTPUT);

pinMode(13, OUTPUT);

Serial.begin(9600); 8. }

void loop() {

incomingdata = Serial.read(); {

if (incomingdata == 'a') {

digitalWrite(2, HIGH);

} else if (incomingdata == 'b') {

digitalWrite(2, LOW);

} else if (incomingdata == 'c') {

digitalWrite(4, HIGH);

} else if (incomingdata == 'd') {

digitalWrite(4, LOW);

} else if (incomingdata == 'e') {

digitalWrite(12, HIGH);

digitalWrite(13, LOW);

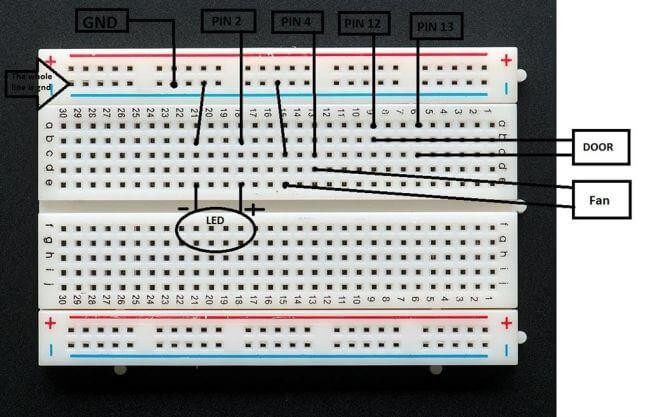
delay(3500);

digitalWrite(12, LOW);

} else if (incomingdata == 'f') {

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| . | |  | |  | digitalWrite(12, LOW); | |
| . | |  | |  | digitalWrite(13, HIGH); | |
| . | |  | |  | delay(3500); | |
| - | |  | |  | digitalWrite(13, LOW); | |
| . | |  | | } |  | |
|  | | } | |  |  | |
|  |  | |  | | |

**Circuit Diagram:**

****

**C# Source Code**:

using System;

using System.Windows.Forms;

namespace HomeAutomationUsingButtons {

public partial class Form1: Form {

public Form1() {

InitializeComponent(); }

private void button1\_Click(object sender, EventArgs e) {

serialPort1.Open();

serialPort1.Write("a");

serialPort1.Close();

}

private void button6\_Click(object sender, EventArgs e) {

serialPort1.Open();

serialPort1.Write("b");

serialPort1.Close();

}

private void button2\_Click(object sender, EventArgs e) {

serialPort1.Open();

serialPort1.Write("c");

serialPort1.Close();

}

private void button5\_Click(object sender, EventArgs e) {

serialPort1.Open();

serialPort1.Write("d");

serialPort1.Close();

}

private void button3\_Click(object sender, EventArgs e) {

serialPort1.Open();

serialPort1.Write("e");

serialPort1.Close();

}

private void button4\_Click(object sender, EventArgs e) {

serialPort1.Open();

serialPort1.Write("f");

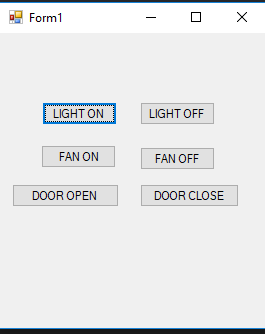
serialPort1.Close();

}

}

. }

**Form Output:**



**10. Servo motor operations with Arduino.**

**Required Components:**

1. ARDUINO UNO
2. Jumper Wires
3. Bread Board
4. Servo motor
5. USB Cable – Used to connect ARDUINO with Computer to send commands.

**Servo motor:**

**Program:**

#include <Servo.h>

int servoPin = 3;

Servo Servo1;

void setup() {

Servo1.attach(servoPin);

}

void loop(){

Servo1.write(0);

delay(1000);

Servo1.write(90);

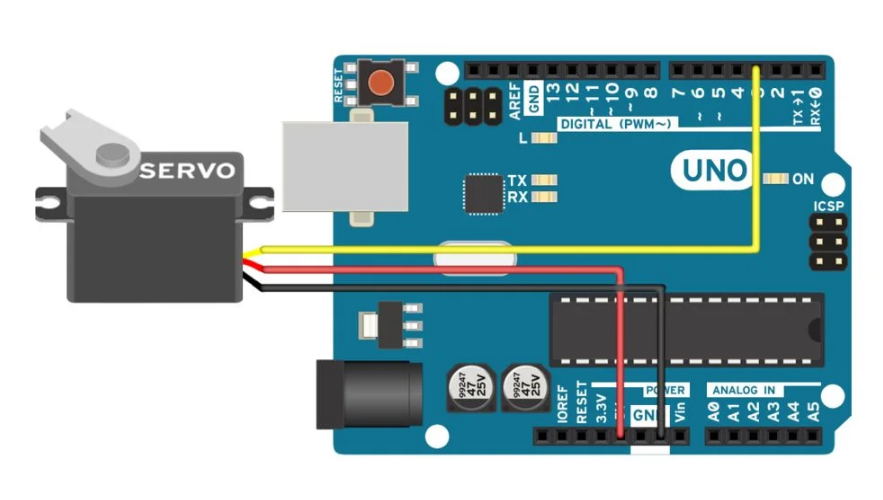
delay(1000);

Servo1.write(180);

delay(1000);

}

**Circuit Diagram:**

****