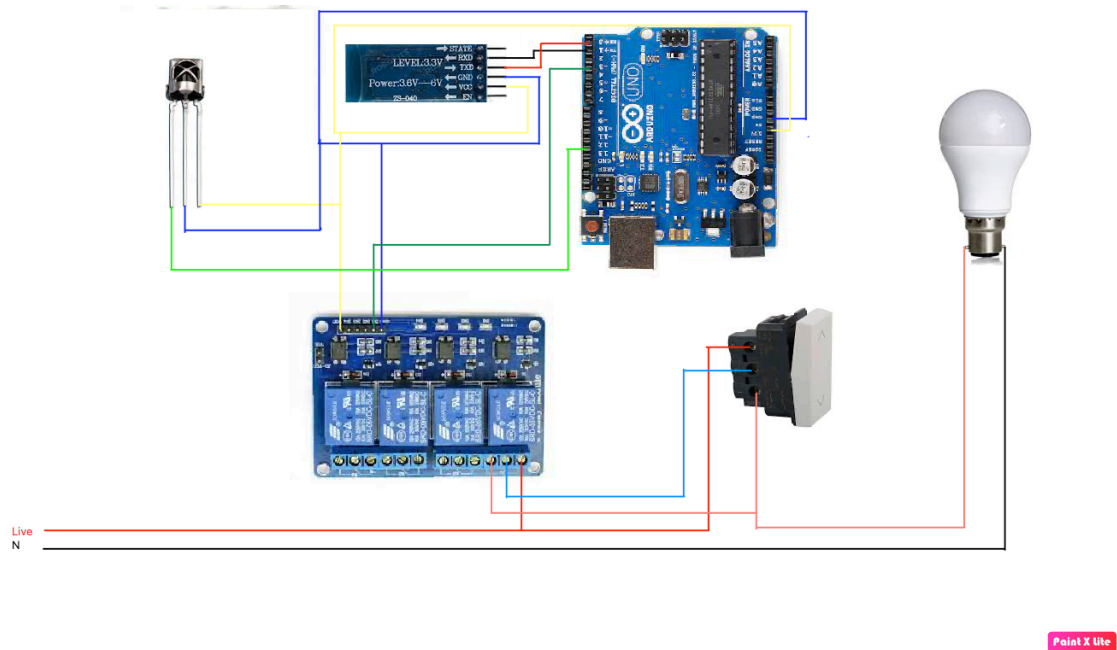


Home Automation using Arduino (Bluetooth & remote control)



Designed By:
Hiten Nandasana

Home Automation using Arduino

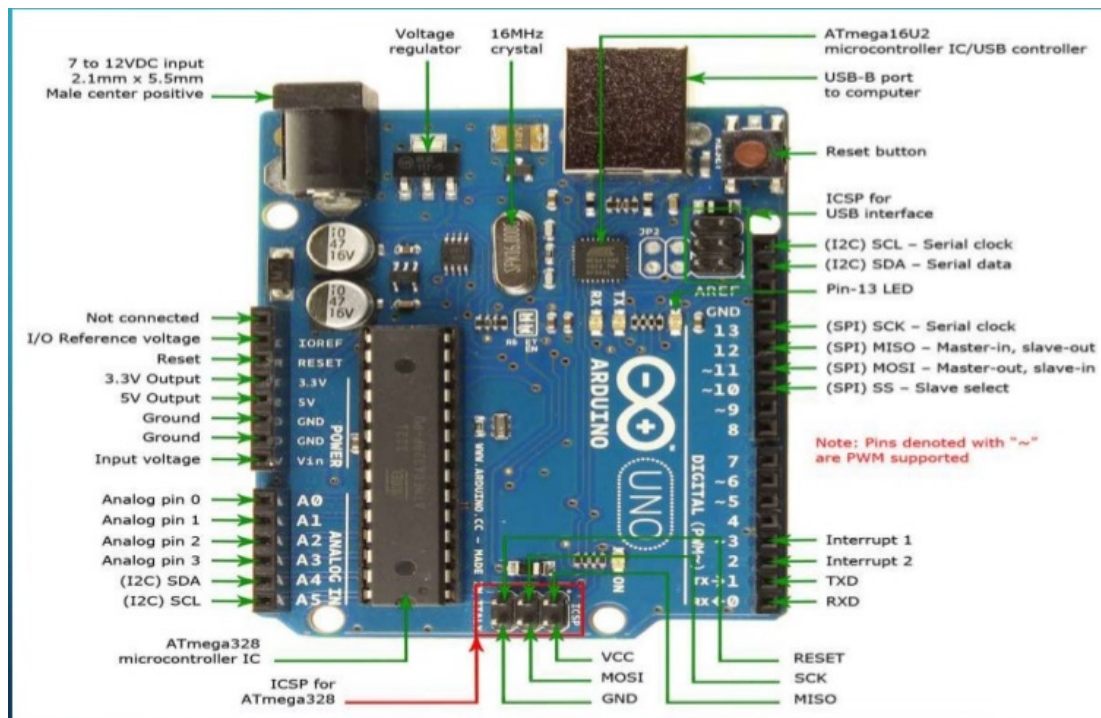
Introduction:

- To develop a Bluetooth based home automation system with Arduino-UNO Board and an Android Application.
- Remote-controlled home automation system provides a simpler solution with Android Application technology.
- Remote operation is achieved by any smart phone/Tablet etc.

Hardware requirements:

- ARDUINO UNO
- BLUETOOTH MODULE – HC-05
- 12V RELAY
- IR RECEIVER
- BREAD BOARD
- JUMPER WIRES
- REMOTE

ARDUINO UNO:



- THE ARDUINO UNO IS A MICROCONTROLLER BOARD ON THE ATMEGA328P.
- IT HAS 14 DIGITAL INPUT/OUTPUT PINS (OF WHICH 6 CAN BE USED AS PWM OUTPUTS), 6 ANALOG INPUTS, A 16 MHZ QUARTZ CRYSTAL, A USB CONNECTION, A POWER JACK, AN ICSP HEADER AND A RESET BUTTON.X
- SIMPLY CONNECT IT TO A COMPUTER WITH A USB CABLE OR POWER IT WITH A AC-TO-DC ADAPTER OR BATTERY TO GET STARTED.

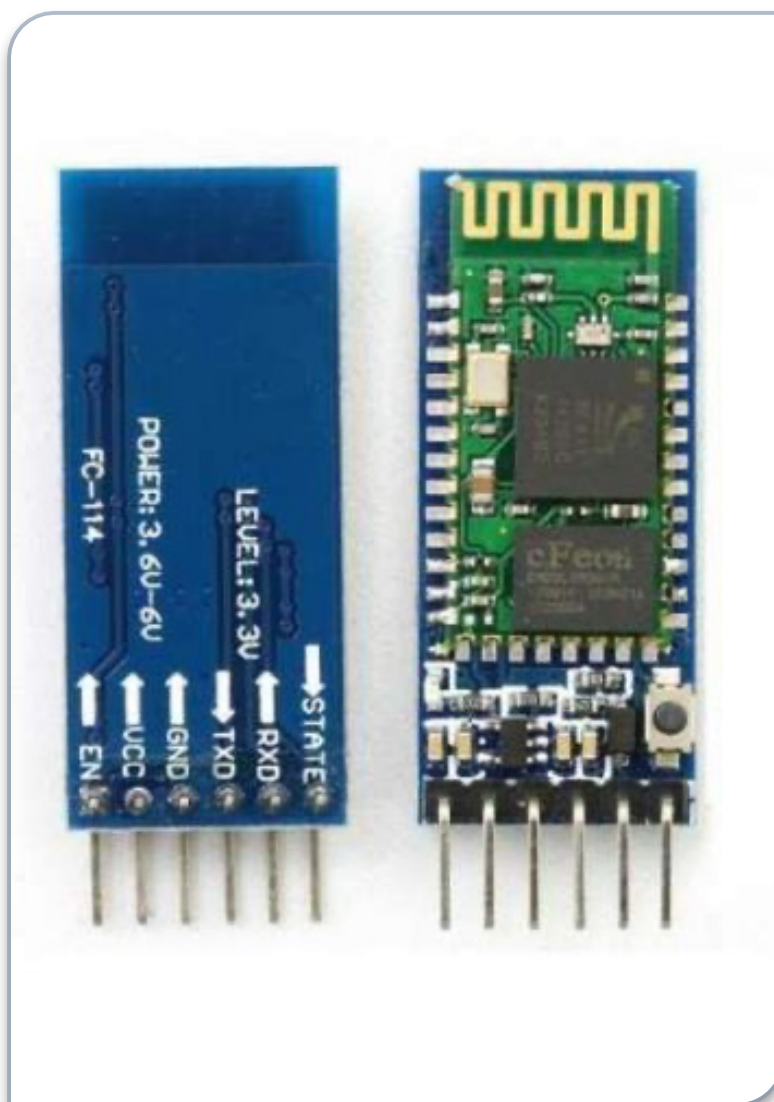
General pin functions:

- **LED:** There is a built-in LED driven by digital pin 13. When the pin is high value, the LED is on, when the pin is low, it is off.
- **VIN:** The input voltage to the Arduino board when it is using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V:** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 20V), the USB connector (5V), or the VIN pin of the board (7-20V). Supplying

voltage via the 5V or 3.3V pins bypasses the regulator, and can damage the board.

- **3V3:** A 3.3-volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND:** Ground pins.
- **Reset:** Typically used to add a reset button to shields that block the one on the board.

Bluetooth module:



- It is used for many applications like wireless headset, game controllers, wireless mouse, wireless keyboard and many more consumer applications.

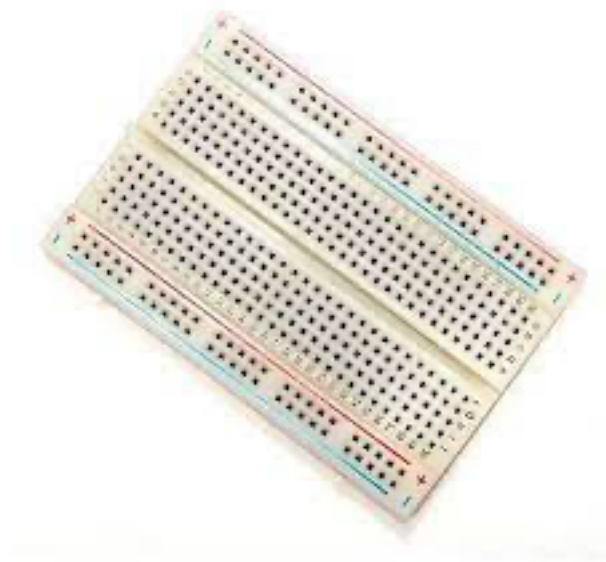
- It has range up to <100m which depends upon transmitter and receiver, atmosphere, geographic & urban conditions.
- It is IEEE 802.15.1 standardized protocol, through which one can build wireless Personal Area Network (PAN). It uses frequency-hopping spread spectrum (FHSS) radio technology to send data over air.
- It uses serial communication to communicate with devices. It communicates with microcontroller using serial port (USART).
- For the communication between mobile phone and Arduino Bluetooth module (HC-05) is used.
- HC-05 is low power 5V operation and is easy to use with Bluetooth SPP (serial port protocol).

Relay module:



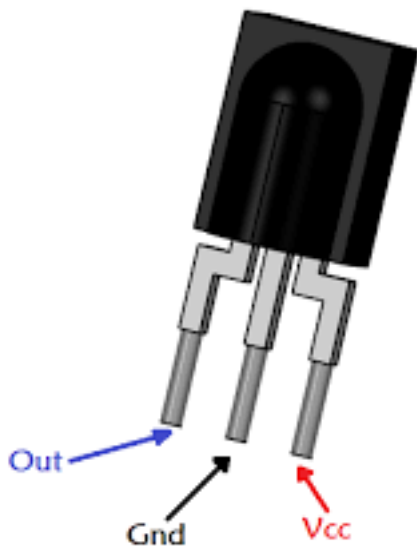
- A relay is an electrically operated switch that can be turned on or off, letting the current go through or not, and can be controlled with low voltages, like the 5V provided by the Arduino pins.
- In this project we used a 12V 4 pin relay.
- Controlling a relay module with the Arduino is as simple as controlling any other output as we'll see later on.
- This relay module has two channels (those blue cubes). There are other models with one, four and eight channels. This module should be powered with 5V, which is appropriate to use with an Arduino. There are other relay modules that are powered using 3.3V, which is ideal for ESP32, ESP8266, and other microcontrollers.

Bread Board:



- A **breadboard** is a solderless device for temporary prototype with electronics and test circuit designs. Most electronic components in electronic circuits can be interconnected by inserting their leads or terminals into the holes and then making connections through wires where appropriate.

IR Receiver:



- The TSOP38238 is an IR Receiver module for infrared remote control systems. In TSOP38238, a PIN diode and a preamplifier are assembled on a lead frame while the epoxy package acts as an IR filter. The operating voltage range for the TSOP38238 IR Receiver diode is from 2.5V to 5.5V.

Jumper Wires:



Working:

- Simply First we connect the Components with Arduino, Relay and Bread Board as shown in Above First Figure.
- Than we Connect the Arduino with the PC and using Arduino Software we install the Below code in Arduino Board.
- Next we connect Arduino with 5V Adapter for Power Supply.
- Now we simply use of Remote And Bluetooth Application on Any Smart Phone as well as Switch we Operate any Devices in Home to ON/OFF.

Code:

```
#include <IRremote.h>
int LED1=3;
int LED2=4;
int LED3=5;
int LED4=6;
int irPin = 12;
```

```
IRrecv irRecv(irPin);
decode_results results;
```

```
int LED1Flag = 0;
int LED2Flag = 0;
int LED3Flag = 0;
int LED4Flag = 0;
```

```
String input;
String login_pass="3791"; // Make sure BT Pin is - 6789
```

```
void setup() {
  Serial.begin(9600);
  IrReceiver.enableIRIn();
  pinMode(LED1, OUTPUT);
```



```

pinMode(LED2, OUTPUT);
pinMode(LED3, OUTPUT);
pinMode(LED4, OUTPUT);
digitalWrite(LED1, LOW);
digitalWrite(LED2, LOW);
digitalWrite(LED3, LOW);
digitalWrite(LED4, LOW);
}

void loop() {

  if (IrReceiver.decode()) {

    Serial.println("Received value = ");
    Serial.println(IrReceiver.decodedIRData.decodedRawData);

    switch (IrReceiver.decodedIRData.decodedRawData) {
      case 1838349569:
        LED1Flag = change_state(LED1 , LED1Flag, "LED 1 ON", "LED 1 OFF");
        break;

      case 1821637889:
        LED2Flag = change_state(LED2 , LED2Flag, "LED 2 ON", "LED 2 OFF");
        break;

      case 869072129:
        LED3Flag = change_state(LED3 , LED3Flag, "LED 3 ON", "LED 3 OFF");
        break;

      case 1905196289:
        LED4Flag = change_state(LED4 , LED4Flag, "LED 4 ON", "LED 4 OFF");
        break;
    }

    irRecive.resume();
  }
  fun2();
}

```

```

int change_state (int LED, int state, String on_msg, String off_msg)
{
  if(state == 0)
  {
    Serial.println(on_msg);
    digitalWrite(LED, HIGH);
    state = 1;

  }
  else
  {
    Serial.println(off_msg);
    digitalWrite(LED, LOW);
    state = 0;
  }
  return state;
}

```

```

void fun2()
{
  if(Serial.available()>0)
  {
    input= Serial.readString();
    if(input=="1")
    {
      LED1Flag = change_state(LED1 , LED1Flag, "LED 1 ON", "LED 1 OFF");
    }
    else if(input=="2")
    {
      LED2Flag = change_state(LED2 , LED2Flag, "LED 2 ON", "LED 2 OFF");
    }
    else if(input=="3")
    {
      LED3Flag = change_state(LED3 , LED3Flag, "LED 3 ON", "LED 3 OFF");
    }
    else if(input=="4")
    {
      LED4Flag = change_state(LED4 , LED4Flag, "LED 4 ON", "LED 4 OFF");
    }
  }
}

```

}

Application:

- Using this project, we can turn on or off appliances remotely i.e. using a phone or tablet.
- The project can be further expanded to a smart home automation system by including some sensors like light sensors, temperature sensors, safety sensors etc. and automatically adjust different parameters like room lighting, air conditioning (room temperature), door locks etc. and transmit the information to our phone.
- Additionally, we can connect to internet and control the home from remote location over internet and also monitor the safety.

Conclusion:

- In this practical we get to learn the Basics of Arduino. We learn how to send a high (ON) and low (OFF) signal to a digital pin and give a time delay to it.