

Arduino - Control LEDs with a Remote Control by RuiSantos



Author: RuiSantos Random Nerd Tutorials

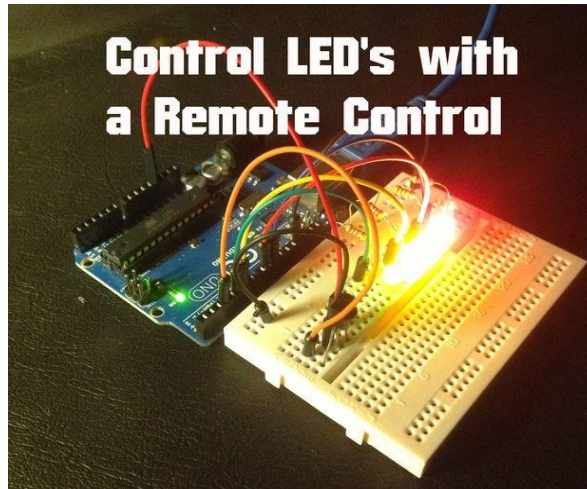
Hi guys, My name is Rui Santos. Who am I? I'm a 19 years old student currently studying at FEUP Electrical Engineering. I'm Portuguese. I've created this account to share my electronics projects. If you guys enjoy my projects you can visit my website for more information. Or contact me by sending an email to rffsantos16 at gmail.com

Intro: Arduino - Control LEDs with a Remote Control

Hi guys,

I'm sharing with you a project I've recently done. It's about how you can control some LED's with a remote control and your Arduino. But you can apply this to any electronic device such as motors, lights for example.

you can visit my website for more electronic projects, interesting news and tips:
<http://randomnerdtutorials.com/>



Step 1: IR library

First you need to go to this page and download the IR library. You just need to follow the read me file to install. I think that any IR remote control can do this, I'll be using a Philips Universal one.

Step 2: Parts required

Parts list:

- 1 x Arduino
- 1 x Breadboard
- 1 x Remote control
- 1 x IR receiver (I'll be using TSOP 4838)
- 4 x LED's
- 4 x 220ohm resistors
- Jumper cables

The infrared receiver has 3 pins:

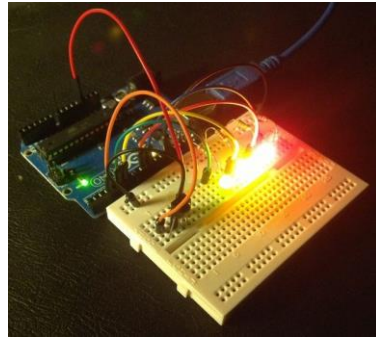
First pin: Vout, outputs HIGH when no signal is present and LOW when a mark is received.

Second pin: GND.

Third pin: Vcc.

For more Projects and Electronics Tips: <http://randomnerdtutorials.com>

Want to see my Projects in action? http://www.youtube.com/subscription_center?add_user=randomnerdtutorials



Step 3: Circuit diagram

You need to wire your circuit something like this.

And then I went to the arduino IDE > file>examples>IRremote> IRrecvDemo. You need to upload the sketch to your arduino, open the serial monitor and start using your remote control and see which values the arduino is receiving.

After a while I've wrote down which values appear in the serial monitor when you press the volume up key or any other key, and write it down for all the keys you want to use. And they were:

Power: E240

Forward: E250

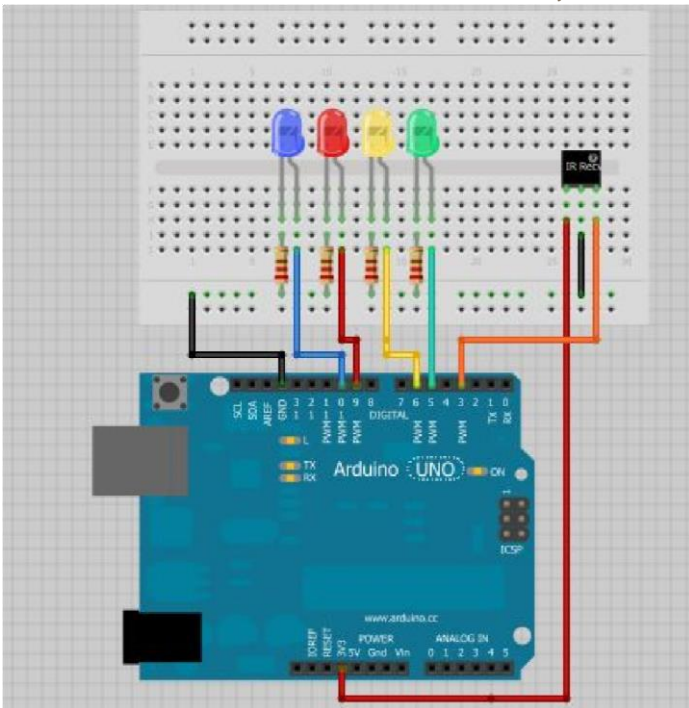
Reverse: E248

Volume+: E244

Volume-: E254

Mute: E24C7

You will need to convert these hexadecimal numbers to decimal, you can use this tool for that.



For more Projects and Electronics Tips: <http://randomnerdtutorials.com>

Want to see my Projects in action? http://www.youtube.com/subscription_center?add_user=randomnerdtutorials

Step 4: Upload the Arduino Code

Upload this code

```
/*
 * IRremote Library - Copyright 2009 Ken Shirriff
 * created by Rui Santos, http://randomnerdtutorials.com
 * Control LED's with a remote control
 * 2013
 */

#include <IRremote.h>

int IR_Recv = 3; //IR Receiver Pin 3
int g_ledPin = 5; //green LED pin 5
int y_ledPin = 6; //yellow LED pin 6
int r_ledPin = 9; //red LED pin 9
int b_ledPin = 10; //blue LED pin 10
int ledPins[] = {5, 6, 9, 10}; //array with all the LED's pins
int ledStates[] = {0, 0, 0, 0}; //this means the LED's states at first is 0 = LOW
int i=0; //LED index for the arrays

IRrecv irrecv(IR_Recv);
decode_results results;

//variables to make the LED blink when selected
int ledState = LOW; // ledState to turn the LED on or off long
long previousMillis = 0; // stores last time LED was updated
long interval = 1000; // interval at which to blink (milliseconds)

void setup(){
  Serial.begin(9600); //starts serial communication
  irrecv.enableIRIn(); // Starts the receiver
  pinMode(g_ledPin, OUTPUT); // sets the digital pin as output
  pinMode(y_ledPin, OUTPUT); // sets the digital pin as output
  pinMode(r_ledPin, OUTPUT); // sets the digital pin as output
  pinMode(b_ledPin, OUTPUT); // sets the digital pin as output
}

void loop(){
  //decodes the infrared input if
  if(irrecv.decode(&results)){
    long int decCode = results.value;
    Serial.println(decCode);
    //switch case to use the selected remote control button
    switch (results.value){
      case 57936: //when you press the Forward button
        //this if/else statement makes sure that LED is ON or OFF before move to the next LED
        if(ledStates[i]==0) digitalWrite(ledPins[i], LOW); else digitalWrite(ledPins[i], HIGH);
        Serial.println("Next LED");
        //makes sure that when we reach the last LED it goes to the first LED again
        if(i==3) i=-1; i+=1; break;

      case 57928: //when you press the Reverse button
        //this if/else statement makes sure that LED is ON or OFF before move to the previous LED
        if(ledStates[i]==0) digitalWrite(ledPins[i], LOW); else digitalWrite(ledPins[i], HIGH);
        Serial.println("Previous LED");
        //makes sure that when we reach the first LED it goes to the last LED
        if(i==0) i=4; i-=1; break;

      case 57932: //when you press the Mute button
        if(ledStates[i]==0){ //if the LED is off, It will turn on
          Serial.println("Turns ON the LED Selected");
          digitalWrite(ledPins[i], HIGH); //sets the LED on
          ledStates[i]=1; //updates the LED state
        }
        else{
          Serial.println("Turns OFF the LED Selected"); //else: the LED is on, It will turn off
          digitalWrite(ledPins[i], LOW); //sets the LED off
          ledStates[i]=0; //updates the LED state
        }
        break;

      case 57920: //when you press the Power button
        Serial.println("Turns OFF all the LED's");
        digitalWrite(g_ledPin, LOW); // sets the green LED off
        ledStates[0] =0; // updates the LED state
    }
  }
}
```

For more Projects and Electronics Tips: <http://randomnerdtutorials.com>

Want to see my Projects in action? http://www.youtube.com/subscription_center?add_user=randomnerdtutorials

```

digitalWrite(y_ledPin, LOW); // sets the yellow LED off
ledStates[1] =0; // updates the LED state
digitalWrite(r_ledPin, LOW); // sets the red LED off
ledStates[2] =0; // updates the LED state
digitalWrite(b_ledPin, LOW); // sets the blue LED off
ledStates[3] =0; // updates the LED state break;

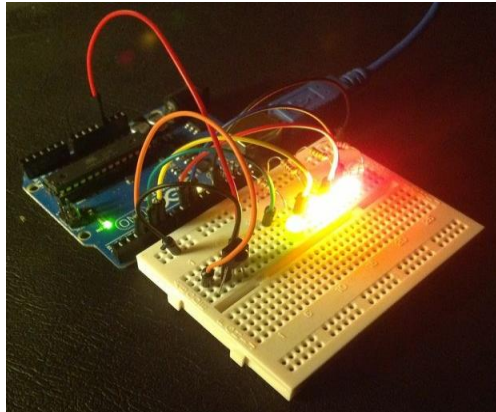
default:
Serial.println("Waiting");
}
irrecv.resume(); // Receives the next value from the button you press
}
//this if statment makes the LED blink if it's selected and off
if(ledStates[i]==0){ unsigned long currentMillis = millis();
if(currentMillis - previousMillis > interval) { // save the last
time you blinked the LED previousMillis = currentMillis;
// if the LED is off turn it on and vice-versa:
if (ledState == LOW)
ledState = HIGH;
else ledState =
LOW;
// set the LED with the ledState of the variable:
digitalWrite(ledPins[i], ledState)
}
}
}
}

```

Step 5: And It's done

For more project and tips you can also check my website: <http://randomnerdtutorials.com/>
You can see this project working right now: <http://www.youtube.com/randomnerdtutorials>

Thanks for Reading!



For more Projects and Electronics Tips: <http://randomnerdtutorials.com>
Want to see my Projects in action? http://www.youtube.com/subscription_center?add_user=randomnerdtutorials