ARDUINO CONTROLLED 12v RGB LED STRIP

by **nathanh1000** on March 13, 2016

Table of Contents

Intro: ARDUINO CONTROLLED 12v RGB LED STRIP

About a year ago I discovered RGB LED strips on Amazon. Being just before the start of music festival season, I naturally went straight to work making crazy bright flashing festival toys. The non-addressable strips being mostly plug and play it was pretty easy to jump right in, but after a while I wanted more control of the lights than I got from the plug and play controllers. I've know for a while that I could use an Arduino to make my own controller, but I hadn't gotten around to it because I thought it would be a fairly complicated electronics project, followed by an equally complicated coding project given that my skill level in both was somewhere between beginner and I'm aware that is a thing.

Not long ago I saw an Instructable from another user showing how to control 12v RGB LED strips with an Arduino N-channel power MOSFETs, and realized that at least electronics side might not be so complicated. I fed my Amazon and ordered these MOSFETs, bookmarked the page and moved Arduino LED controller up a few spots of my to do list.

LIST OF STUFF

Arduino of any persuasion. I used an Uno, but as long as it has at least three PWMs and you know which they are your good.

breadboard or blank pcb or just some electrical tape to prevent shorts. It really is that simple.

wire color coded helps, and if you use the male to female breadboard jumpers the female ends fit right on the MOSFET pins

MOSFETs you need one for each color, and they need be rated to handle the max current you may use. About 330 mA per meter for each channel, 1.66 A per channel for a 5 meter strip. The MOSFETs in the link above are rated for 49 A. Three of those could handle a 150 meters of LEDs. Not that I need it, but for $10 I figured better to be able to cover every inch of my house with LEDs and not need to, than need to and not be able to. FYI: 150 meters would need a 742 A power supply.

RGB LED strip and 12v power supply one Amp per meter of LEDs.
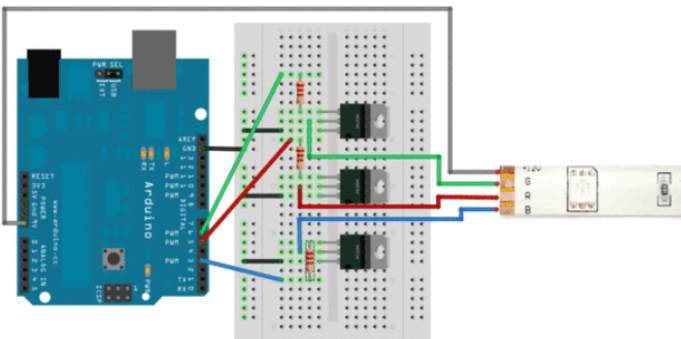
Step 1: Connect your MOSFETs to your Arduino

A week or two later after the MOSFETs arrived, and I had some time to get started, I went back to the bookmarked page, only to have Google tell me it no longer existed. I panicked for a minute at the thought working from memory and then a little more at the thought of screwing up and frying my Arduino...again. My panic was short lived when a quick web search for MOSFET Arduino RGB LED controller brought me to this page. I always forget to check Adafruit, or even the Arduino page.

Adafuit does such a great job explaining it, and the pictures do even better. I think the only thing I can say that you can't get from the pictures is I didn't notice any real difference with or without the resistors. I would imagine the longer the strip you have the more necessary they might be. Other than that Adafruit says something weird about connecting power and ground if you power the LEDs directly, but you only need to connect the grounds. Be very care where you connect 12v on the Arduino. Actually connecting 12v anywhere but the DC power socket will pretty much fry it instantly. vin should be ok to, but why connect power a mm away from an almost identical pin that would basic be the self destruct button if you connected to it. That's like playing that stabby knife finger torture game. You're scaring the hell out of your fingers for no reason, and there's a very real possibility you're gonna loose one of them eventually.
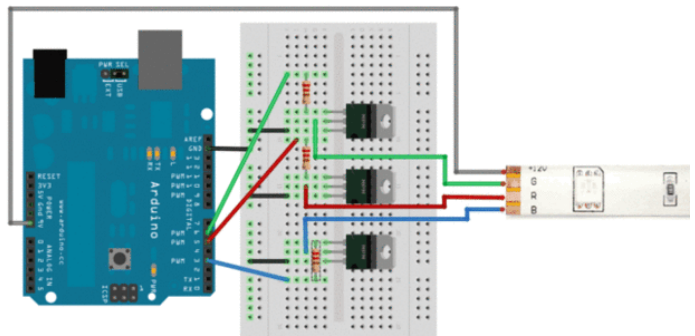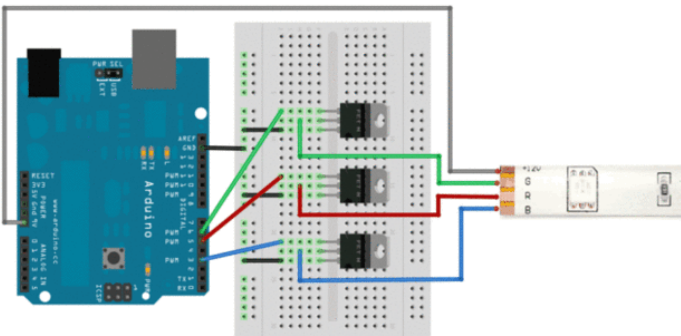
Step 2: Code

Besides the fact the orginal instructable I saw is gone, the code for the Arduino is the real reason I wanted to make this instructable. Adafruit has a simple fade code for the Arduino in the link, and it's a good start to understanding how the Arduino controls the lights, but it's pretty much what the plug and play controllers do and there's not much room for altering. The altering you can do is extremely tedious. I searched for other example codes, and found tons of code for digital strips, and a couple of altered forms of the code in the Adafruit link. So I spent the weekend reading parts (short attention span) of tutorial C++ and getting little lessons from my amateur programmer roommate. He wrote a code for me, attached file Jacobs_rolling_color_fade, which is pretty cool, and can be altered for lots of cool variations. After playing around with his code for a day and getting a feel for how the C++ language works I wrote my own code, attached file My_Color_Fade.

Pretty proud of my accomplishment I took it to my amateur programmer roommate, who is also a full time bubble buster, who did not have the same feelings of admiration for my code that I had. The best he could say was "it does what you want it to do."

But that's exactly the point. It may not do it the best way from a programmer's perspective, but it does exactly what I wanted it to do, and writing, observing and tweeking it helped me to learn a lot about how to code for Arduino, and definitely how to debug my code. And now festival season is gonna kick ass this year.

One more link Arduino Playground Sketchlist user contributed sketches for all kinds of Arduino projects.

File Downloads

Jacobs_rolling_color_fade.ino (3 KB)
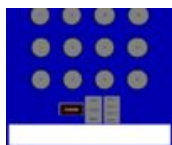[NOTE: When saving, if you see .tmp as the file ext, rename it to 'Jacobs_rolling_color_fade.ino']

My_Color_Fade.ino (5 KB)
[NOTE: When saving, if you see .tmp as the file ext, rename it to 'My_Color_Fade.ino']
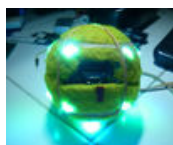
Related Instructables

RGB LED controled with arduino by FazilM2

LED Cube Code Generator 4X4X4 RGB by BoucherM

Simple LED strip party lights using arduino by vamsikurre

RGB LED Tennis Ball by bneo99

Arduino Traffic Light for Beginners by joshdont

Arduino Powerd, RGB + White LED, Bluetooth controllable, Floor lamp by bram2202

Comments