adafruit learning system

WiFi Controlled LED Christmahanukwanzaa Tree

Created by Tony DiCola

Last updated on 2020-10-17 01:06:43 AM EDT

Overview

We no longer support CC3000 WiFi modules so this code is for educational uses only!

Decorating a holiday tree with a strip of NeoPixel addressable RGB LEDs (http://adafru.it/1460) is a great way to get into the holiday spirit. Don't settle for a simple fixed color or animation of the LEDs though, expose control of the lights through a web page to control them from a phone, tablet, or computer. You can even let your friends and family control the lights for maximum holiday entertainment!

This project will show you how to use an Arduino Yun (http://adafru.it/1498) or a CC3000 WiFi chip (http://adafru.it/1469) & regular Arduino to control a strip of NeoPixels through a web page. This project is a good example of interacting with an Arduino through a web page that you can use in your own Arduino projects.

Before you begin it will help to familiarize yourself with the following guides:

- NeoPixel Uberguide (https://adafru.it/cEz)
- CC3000 guide (https://adafru.it/cRT) or Arduino Yun overview (https://adafru.it/d1L) (from the main Arduino website)

Continue on to learn about the hardware in this project.

Hardware

Parts

You will need the following parts for this project:

- Christmas tree or other item to decorate. I used a 3 foot tall artificial Christmas tree.
- Strip of NeoPixel LEDs (http://adafru.it/1460). I found 3 meters of the 30 NeoPixel strip worked well to loosely cover my 3 foot tall tree. You can use other NeoPixel products too, like rings (http://adafru.it/1463) or individual pixels (http://adafru.it/1312).
- 5 volt power supply that can supply enough current for all the lights. Remember each pixel can consume up to 60mA, so you will need a large power supply. I used the 5 volt 10 amp supply (http://adafru.it/658) because my strip of 90 pixels can consume over 5 amps of current!
- Large capacitor (http://adafru.it/1589) (1000uF 6.3 volts or higher), as suggested in the NeoPixel guide (https://adafru.it/d1W).
- An Arduino Yun (http://adafru.it/1498), or a CC3000 (http://adafru.it/1469) & Arduino Uno (http://adafru.it/50) / Mega (http://adafru.it/191) / Nano.
- Adapter (http://adafru.it/368) and wires (http://adafru.it/824) to connect the lights, Arduino, and power supply together.

Assembly

If necessary, solder the connector for the NeoPixel strip to the input side of the strip (as indicated by arrows on the strip).

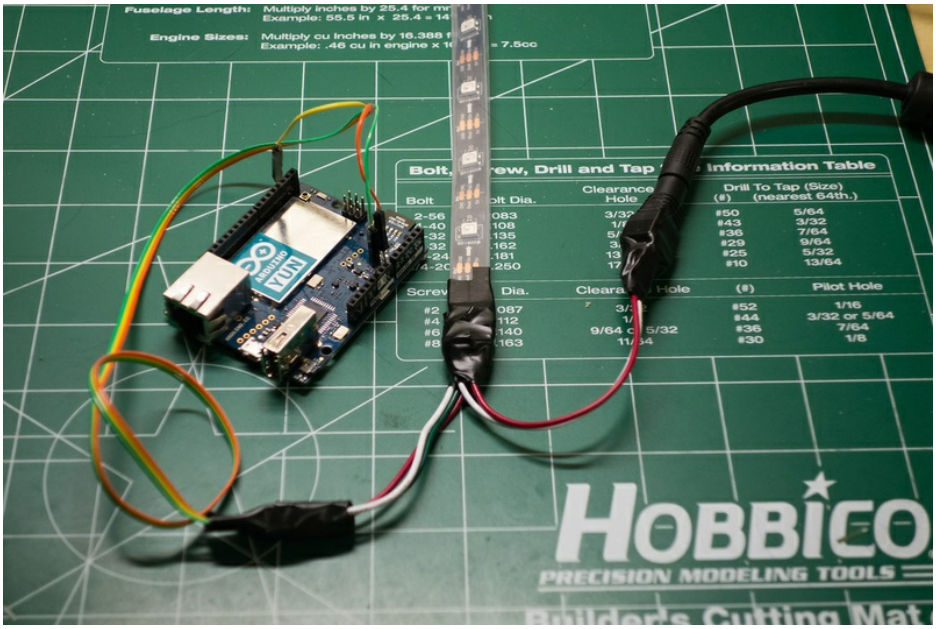
Next connect the strip power and ground wires to the power supply adapter. Add the large capacitor across the power and ground.

Finally connect the input, power, and ground lines from the strip connector to the Arduino. For the input line you can use any free digital pin on the Arduino. If using an Arduino without a 5V regulator, like the Yun, connect power to the 5V or VIN pin and ground to a ground pin. If using an Arduino with a 5V regulator, such as the Uno, it's safest to apply the 5V power to the USB input of the Arduino instead of the 5V pin--you can cut up a spare cable or use this adapter (http://adafru.it/988) to feed in power.

If using a CC3000 and regular Arduino, connect the CC3000 to the Arduino in the same way as other CC3000 examples:

- Arduino 5V to CC3000 VIN
- Arduino ground to CC3000 ground
- Arduino digital pin 13 to CC3000 CLK
- Arduino digital pin 12 to CC3000 MISO
- Arduino digital pin 11 to CC3000 MOSI
- Arduino digital pin 10 to CC3000 CS
- Arduino digital pin 5 to CC3000 VBEN
- Arduino digital pin 3 to CC3000 IRQ

I chose to hide my Arduino in a small cardboard box placed under the tree skirt. I also wrapped connections in black electrical tape to keep my curious cat from getting to them. You can see my hardware connected to an Arduino Yun below:



Be careful decorating your tree with too many lights. The more current your lights require, the more heat they will generate. Don't leave your decorations powered on and unattended!

Continue on to learn about the software in this project.

Software

Download the software for this guide from the following link:

https://adafru.it/d1X

https://adafru.it/d1X

Unzip the archive and you will find 2 Arduino sketches (one for the Yun, and the other for the CC3000), and a webpage directory.

Dependencies

Before going on you will need to install the following dependencies:

- Adafruit NeoPixel library (https://adafru.it/aZU)
- If using the CC3000:
    - Adafruit CC3000 library (https://adafru.it/cFn)
    - CC3000 MDNS library (https://adafru.it/d1Y)

Next, load the appropriate Arduino sketch for your hardware (either the Yun or CC3000 version) in the Arduino IDE. Adjust the define values at the top of the sketch to change the configuration:

- **PIXEL_PIN**: The digital output pin which is connected to the input of the NeoPixel strip.
- **PIXEL_COUNT**: The number of pixels in the NeoPixel strip.

If using the Arduino Yun you will want to make sure it's configured to access your wifi network. Follow the Yun getting started guide (https://adafru.it/d1Z) to connect the Yun to your network.

If using the CC3000, adjust the defines at the top of the CC3000 sketch to configure it to access your wifi network:

- **ADAFRUIT_CC3000_IRQ, ADAFRUIT_CC3000_VBAT, ADAFRUIT_CC3000_CS:** Set these to the appropriate pins which are connected to the IRQ, VBEN, and CS pins respectively on the CC3000.
- **WLAN_SSID, WLAN_PASS, WLAN_SECURITY:** Set these to the appropriate values for your wireless network.

Upload the sketch to your hardware--you should see the strip of LEDs turn on to the default plain light pattern.

Server

This project works by using a computer on your network to host a small control web page. The web page uses Javascript to call a simple REST API on the Arduino which controls the lights. To set up the server you will need to install the following software:

- Python (https://adafru.it/d20) (this project has been tested with Python 2.x, but should work on 3.x)
- Flask (https://adafru.it/d21) (a simple python web framework)
    - The easiest way to install flask is through pip (https://adafru.it/d22), with the command pip install flask (on Mac OSX or Linux run with the sudo command as sudo pip install flask).
- Optionally on Windows, install Bonjour with the Bonjour Printer Services download (https://adafru.it/d23). Bonjour's multicast DNS support is used to simplify configuration by automatically finding the IP address of the Arduino. If Bonjour is not available you can manually configure the Arduino IP address in the web page source.

In a command window, navigate to the webpage subdirectory of the software download. Execute the following command:

```
python server.py
```

A simple web server should start hosting the control webpage. You can access this web page from http://localhost:5000/ () on the server, or http://(your_server_IP_address):5000/ (https://adafru.it/CcY) from another device on your network.

If you see an error that the Arduino IP address could not be found, this means the multicast DNS lookup failed to find the Arduino. Open server.py in a text editor and make sure the value of the **ARDUINO_MDNS_NAME** variable at the top matches the name assigned to your Arduino. On the Arduino Yun this value should be the name you assigned to the Yun in its configuration, with the '.local' suffix added--by default the Yun should use 'arduino.local'. On the CC3000, this value is 'arduino.local' by default (but can be changed in the setup function of the sketch).

If the Arduino MDNS name is correct and still cannot be resolved (for example if Bonjour is not installed on Windows) you can manually find the IP address of the Arduino (check your router's device list) and assign it to the **ARDUINO_IP** variable at the top of server.py. This will override the MDNS check with the IP value you provide. Note that the IP of your Arduino might change when it's restarted.

Once the server is running, access the web page from a browser and start controlling the lights!

Note: If you're using the Yun and have a password protecting the device you will need to enter the Yun's password in the dialog that pops up the first time you press a button on the control page (use username 'root' if prompted too).

Continue on to learn about enhancements you can make to this project.

Future Work

This project is a great example of how to control an Arduino from a web page. By using an Arduino Yun or CC3000 you can extend control of your Arduino to your wireless network.

Some ways to extend this project include:

- Add servos to move and animate other holiday decorations.
- Connect a microphone to make the lights react to sound (https://adafru.it/ciF).
- Add a motion sensor to make the lights react to movement (https://adafru.it/c2F).
- Add more color schemes to the project. The gradient and bar drawing code is very general and supports an arbitrary pattern of colors--look at how the included color schemes are defined by an array of colors to see how to add your own.
- For the Arduino Yun, look at hosting the control web application directly on the Yun's Linux-based processor.

Have fun with the project, and have a happy holiday!