

Partycat Light Controller Kit

By Luke Cyca

Nov 26, 2016

Revision 3

Congratulations on your purchase of the partycat light controller kit! This board uses a ESP8266 wifi-enabled microcontroller to control a string of WS2812-compatible RGB LEDs, plus another channel of high-current PWM dimming. Have fun!

Building

All components are on one side, so it's easy to populate, easy to bake, and easy to tape or hot-glue the finished board down with your LED strip.

R1: 10k Ω Resistor (blue)

R2: 332 Ω Resistor (pink)

R3: 10k Ω Resistor (blue)

R4: 10k Ω Resistor (blue)

R5: 10k Ω Resistor (blue)

C1: 100 μ F 25V Electrolytic Capacitor

C2: 0.1 μ F Ceramic Capacitor (yellow)

U1: AP1117E33G-13

U2: ESP8266 Module 7/12/12E/12F

U3: SN74AHCT125

D1: Red LED

Q2: STB24NF10 MOSFET

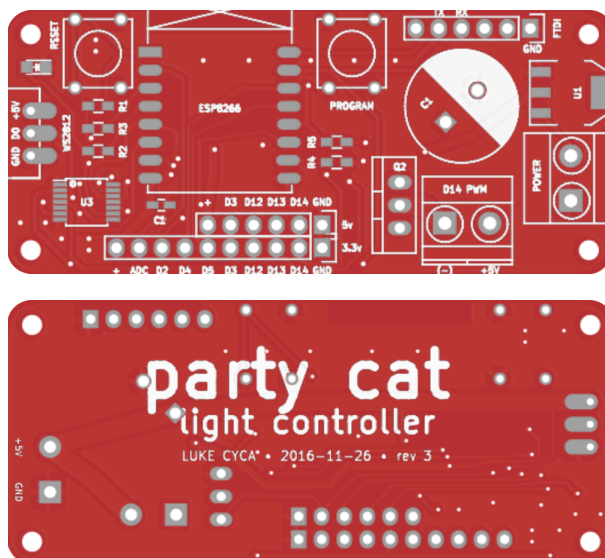
MISC:

2x Pushbuttons

2x Two-terminal screw terminal blocks

0.1" pin headers

A short string of two WS2812 LEDs (included so you can test your board)



SMD Hand-Soldering Pro Tips

Start with the smallest components first.

For each component, put a bit of solder on one pad first. Then while keeping the solder hot with the soldering iron in one hand, place the component with tweezers to tack it in place. Solder a pin on the opposite side. Then go back and resolder the first, and all additional pins.

Use extra solder flux to help the solder flow freely.

Programming

To program this board you need a FTDI cable with 3.3v logic level, like this one from Sparkfun:

<https://new.sparkfun.com/products/9717>

The black wire is GND. Make sure you plug it in the right way around. Some other FTDI cables have a different pinout. Make sure yours is the same as the Sparkfun one:

RTS, RX, TX, 5V, CTS, GND



You can program it using the Arduino IDE along with the ESP8266 board add-on from:

<http://github.com/esp8266/Arduino>

To put the partycat into programming mode, hold down the PROGRAM button while pressing the RESET button. Then upload your sketch.

This board is set up to use the i²c output (pin 3) to control the lights with DMA, allowing relatively high frame rates without completely tying up the processor. Use will need WS2812 i²c library: https://github.com/JoDaNI/esp8266_ws2812_i2s

Here is an example sketch you can use to test your board:

<https://github.com/lukecyca/partycat-blinky>

The partycat is also compatible with other NeoPixel libraries including NeoPixelBus:

<https://github.com/Makuna/NeoPixelBus>

Since pin 3 is also used by the ESP8266 for UART RX, **unplug the FTDI cable after programming for best results driving the WS2812s.**

Other Features

Some GPIO pins are broken out with both 3.3v and 5v variants. The 3.3v one is directly connected to the ESP8266 pin so it can be used as either input or output. The corresponding 5v variant pin can only be used as an output.

There is a high-current-sinking MOSFET connected to PIN D14 that can be used to drive a string of regular LEDs, a motor, etc. I have used this for driving a string of white LEDs to reinforce the WS2812s when I want a very bright white. Since it is sinking and switches the low side, it can be used to drive higher voltage (ex: 12v) LED strips.



Schematic

