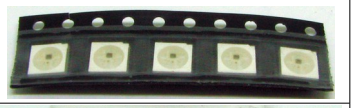
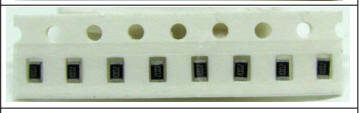
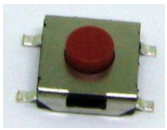
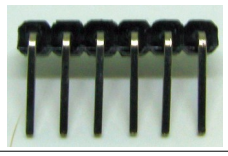
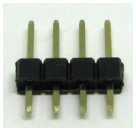
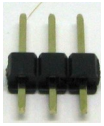
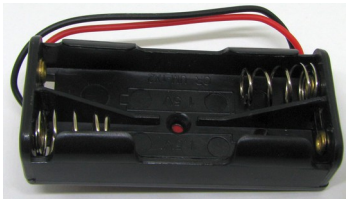
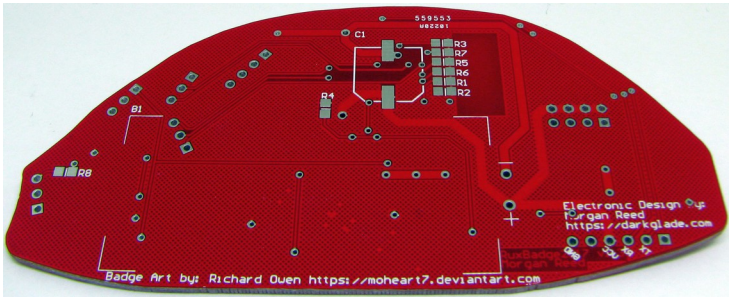
Ruxcon Hardware Hacking Village 2017

RuxBadge2017 instruction sheet

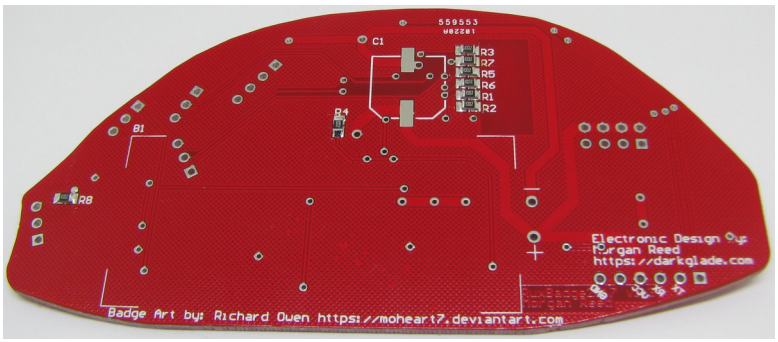
Bill of materials;

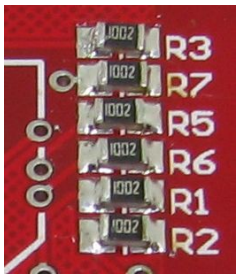
Qty	Description	Designators	Image
1	ESP-12E Module	U1	
1	1000uF 16V SMD Electrolytic	C1	
5	WS2812B RGB LED	LED1-5	
8	10k Resistor (0805 SMD)	R1-8	
6	SMD Tactile Switch	S1-6	
1	6-way Right Angle Header	P1	
1	4x2 Header	P2	
1	4-way Header	P3	
4	3-way Header	P4-7	
1	2xAAA Battery Holder	B1	

These instructions will be a lot less verbose than the ones for SimpleSolder as it's expected that you've probably grasped the basics. If you have any issues feel free to talk to one of the HHV staff.

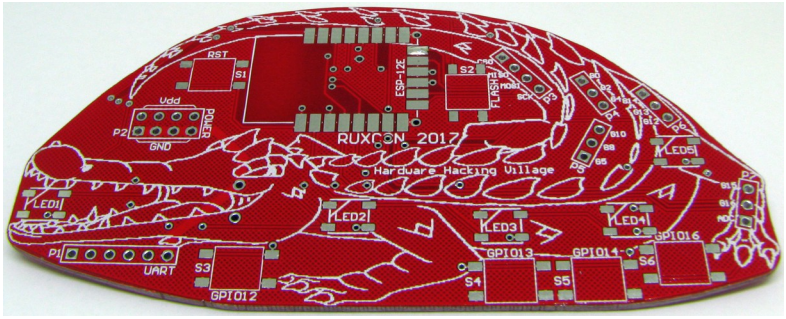


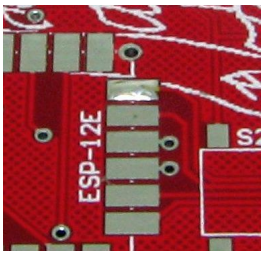
0. We are going to be mounting parts on BOTH sides of the board for this project, so we'll start with the eight SMD resistors on the back side.



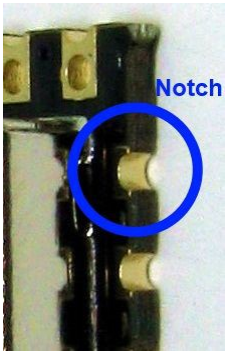


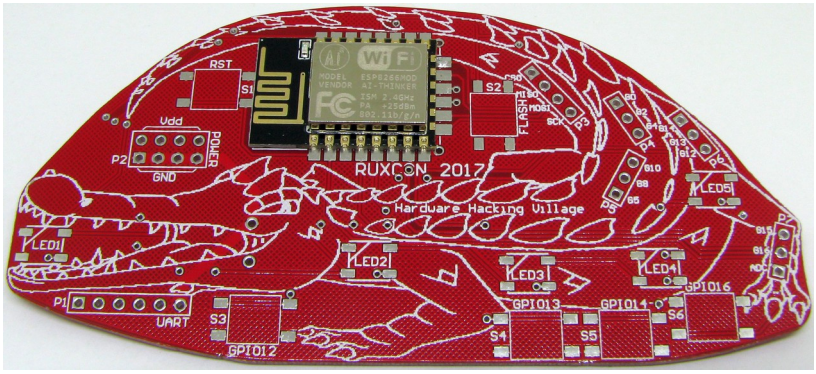
1. All of the resistors are the same value so you don't need to worry about getting the right bits into the right places. Note above that all of the resistors are oriented (Or “dressed”) such that they all read the same way (Barring R4 since it's vertical).

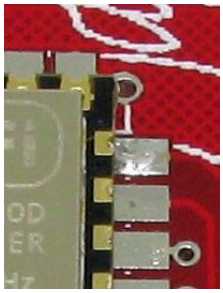




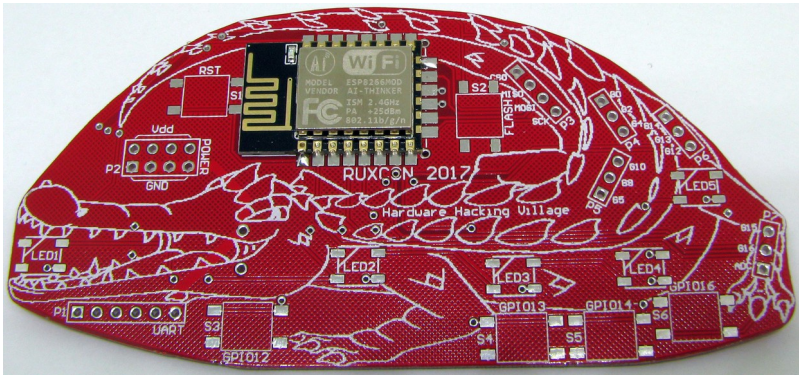
2. Next up we'll mount the ESP-12E module, the connections are quite far apart but soldering castellated modules can be knacky. The only real trick is making sure you're heating both the pad on the board and the “notch” on the module. First off you want to tin one of the pads for the ESP-12E, as pictured above.

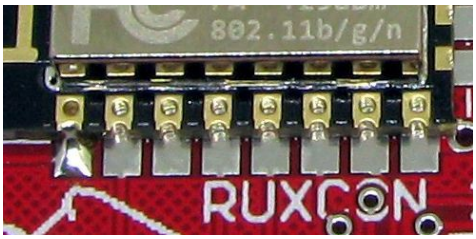




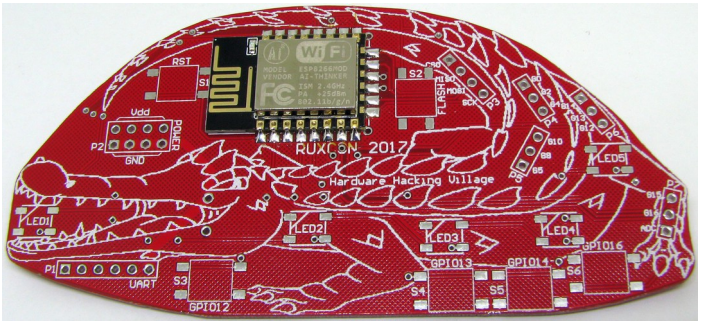


3. Place the module over the footprint and heat both the tinned pad and the “notch” on the module. The solder should wick into the “notch” and connect the two points, if it does not try adding a little more solder and ensure you're touching both the pad on the board and the “notch” on your module with your soldering iron. You don't need to worry about getting the module perfectly aligned at this point, the important thing is to establish the connection.

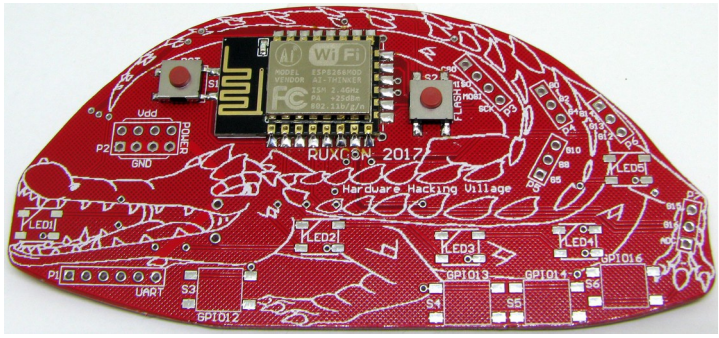


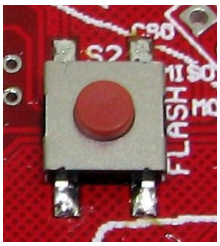


4. Now that we've got the module tacked in place we can worry about alignment, heat the pad you soldered and tweak the module such that it lines up with the pads on the board, when you are happy that all the pads are lined up, solder another pad on the device (Apply your soldering iron such that it's touching both the pad on the board and the “notch” on the module, then feed in solder.)

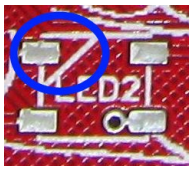


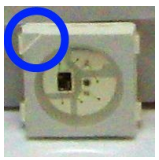
5. Go through and solder the rest of the pads on the module.

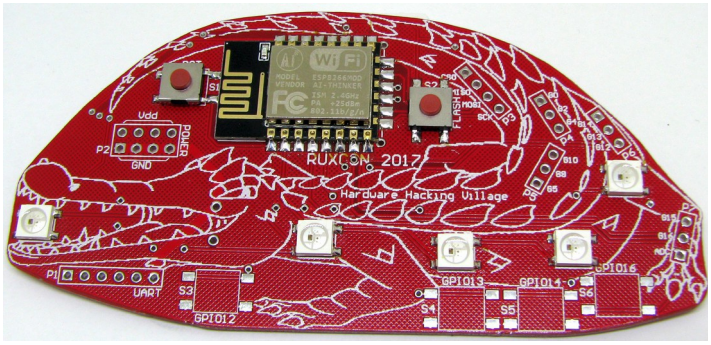




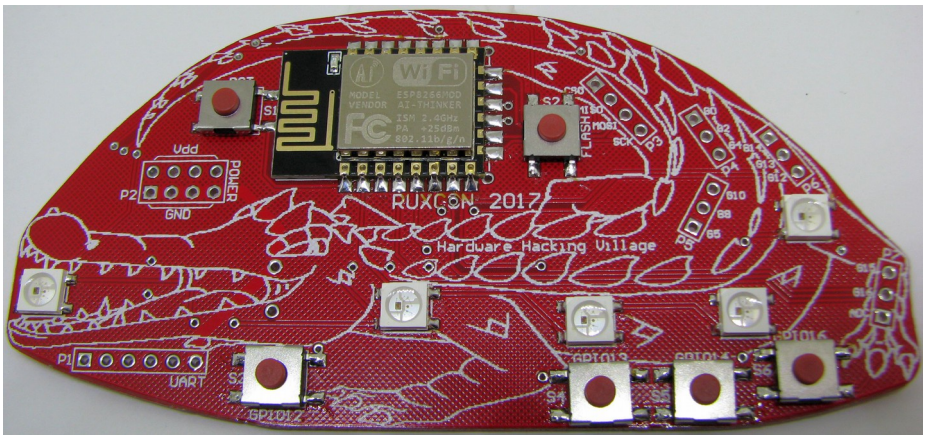
6. Next up is the two push button switches towards the top of the board. For these guys, tin one pad, tack the switch down, make sure it aligns with the other three pads (Note that the legs of the buttons "ride" the outside edge of the pads), then solder the rest of the connections and touch up the first if needs be.



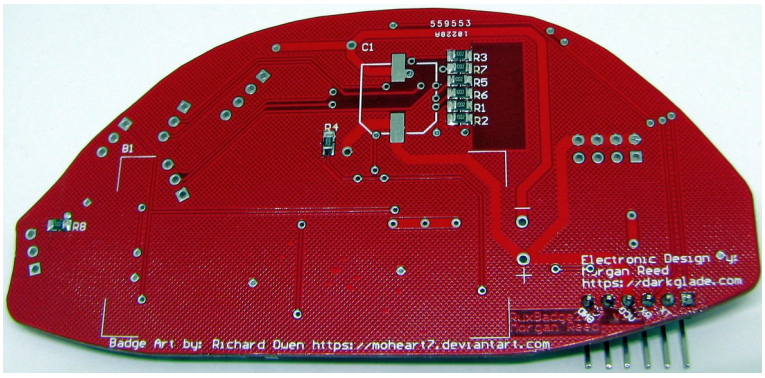




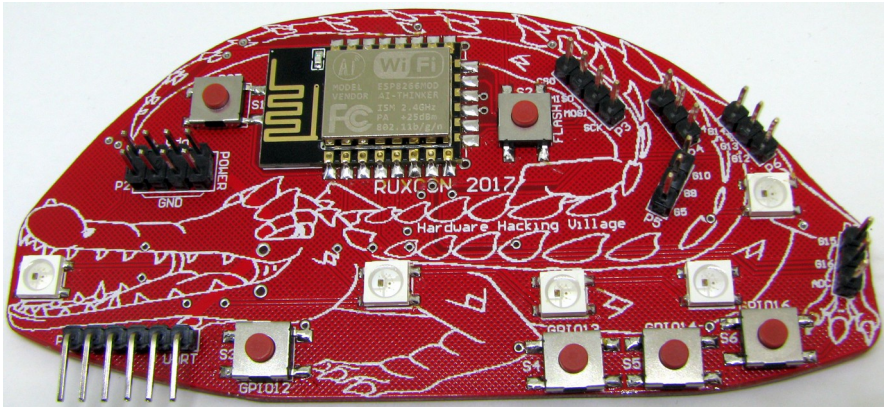
7. Now for the LEDs, notice that one corner of the LEDs have a "chamfer", this chamfer lines up with the angled line on the PCB footprint. Same story as elsewhere here, tin one pad, tack part in place, tweak as needed, solder other three connections.



8. Next we fit the other four buttons. Repeat as for step 6 above. You'll need to take a bit of extra care with the top left pad of S6, as it's very close to LED4. If you touch the LED's body with the soldering iron, it will melt a bit. Most likely it will still work but melted components are ugly so take care ;)

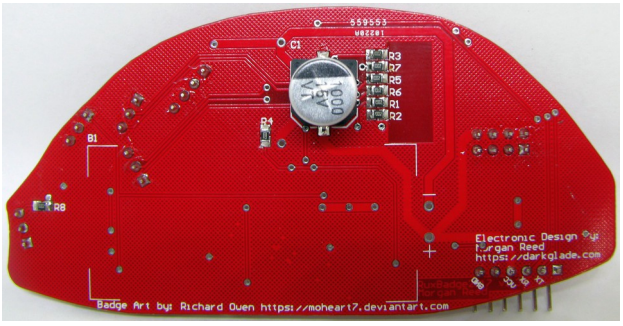


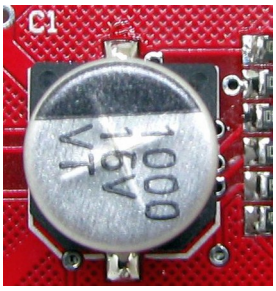
9. Next, the 6-way UART header. The “short” part of the pins passes through from the top of the board, solder one of the pins, then adjust the alignment if necessary and solder the rest of them.



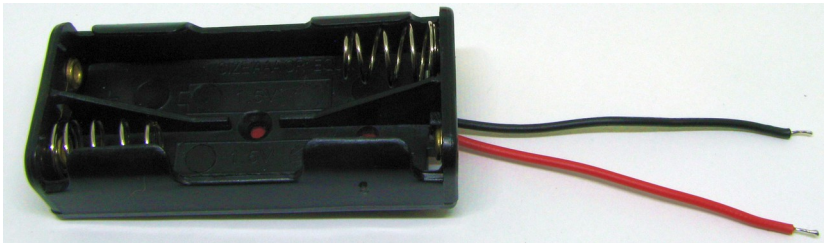
10. (OPTIONAL) The breakout headers are basically there to allow you to easily connect stuff to the ESP module on the badge, they are not essential for the function of the badge and can be omitted, it will also fit better in the pouch on your lanyard if you omit them :). P5 is the only one which can't readily be fitted “after the fact” (Due to the battery holder being in the way.) but if you needed to you could solder wires into the holes from the top of the board.

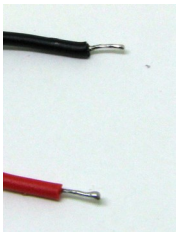
Follow the same process as for the UART header in step 9 for these if you want to install them.



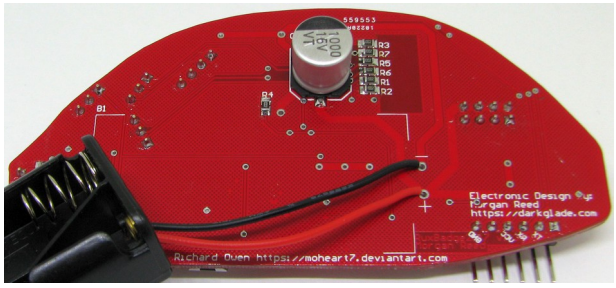


11. Flip your board over, it's time to mount the main filter cap, the black plastic part on the bottom lines up with the marking on the PCB, same deal as everything else, tin pad, tack part, align part, solder other lead.





12. Last part now, the battery holder (I'd have preferred to use a PCB mount battery holder here but nobody on eBay seems to sell them in PCB mount). The leads on the battery holder are WAY longer than they need to be, trim them to about 5cm (You can get away with shorter but it makes it fiddlier), strip the ends and “tin” the wires (Apply a small amount of solder to them to make them “solid”).



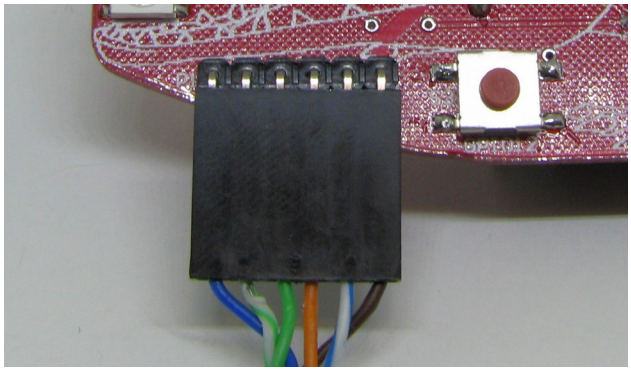
13. Insert the wires of the battery holder into the two pads, the Red goes to the + pad Black goes to the – pad, flip the board over and solder both wires in place, once it's soldered, peel the backing off the double-sided tape on the back of the holder and stick it in the area indicated on the board. If you fitted P5 earlier you may need to trim the pins to get the holder to fit snugly against the board.

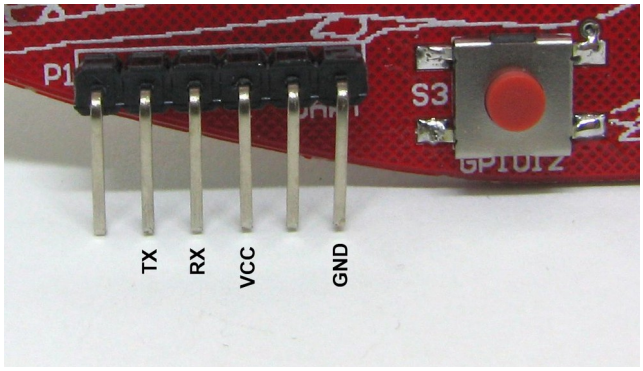
Et Voila! Build complete. Now proceed onto the next section for how to load the badge firmware.

Loading Firmware

The firmware can be downloaded from https://ruxconhhv.darkglade.com/2017/ there are two flavours there. The “CTF” version which includes the badge flag (Some of the modules are pre-compiled), the other is pure lua source without the flag functionality in it.

The first thing you'll need to upload the firmware is a 3.3V USB-UART adapter (MUST be 3.3V, the ESP8266 is NOT 5V tolerant), if you don't have one talk to the HHV staff, we have several available for loan.





IMPORTANT: Remove the batteries before connecting to your USB-UART adapter.

If you're using one of the loaner adapters, plug them in such that the brown wire is on the innermost pin of the UART header (Left hand photo above), if you're using your own connect it up per the diagram on the right (The pinout is compatible with the defacto standard “FTDI” pinout but does not support auto-reset functionality).

Connect the badge to the adapter then connect the adapter to your PC, hot plugging the badges generally results in a power surge that causes the USB port to shut down.

Loading the firmware is the same in either case (Assuming you're running some flavour of Linux);

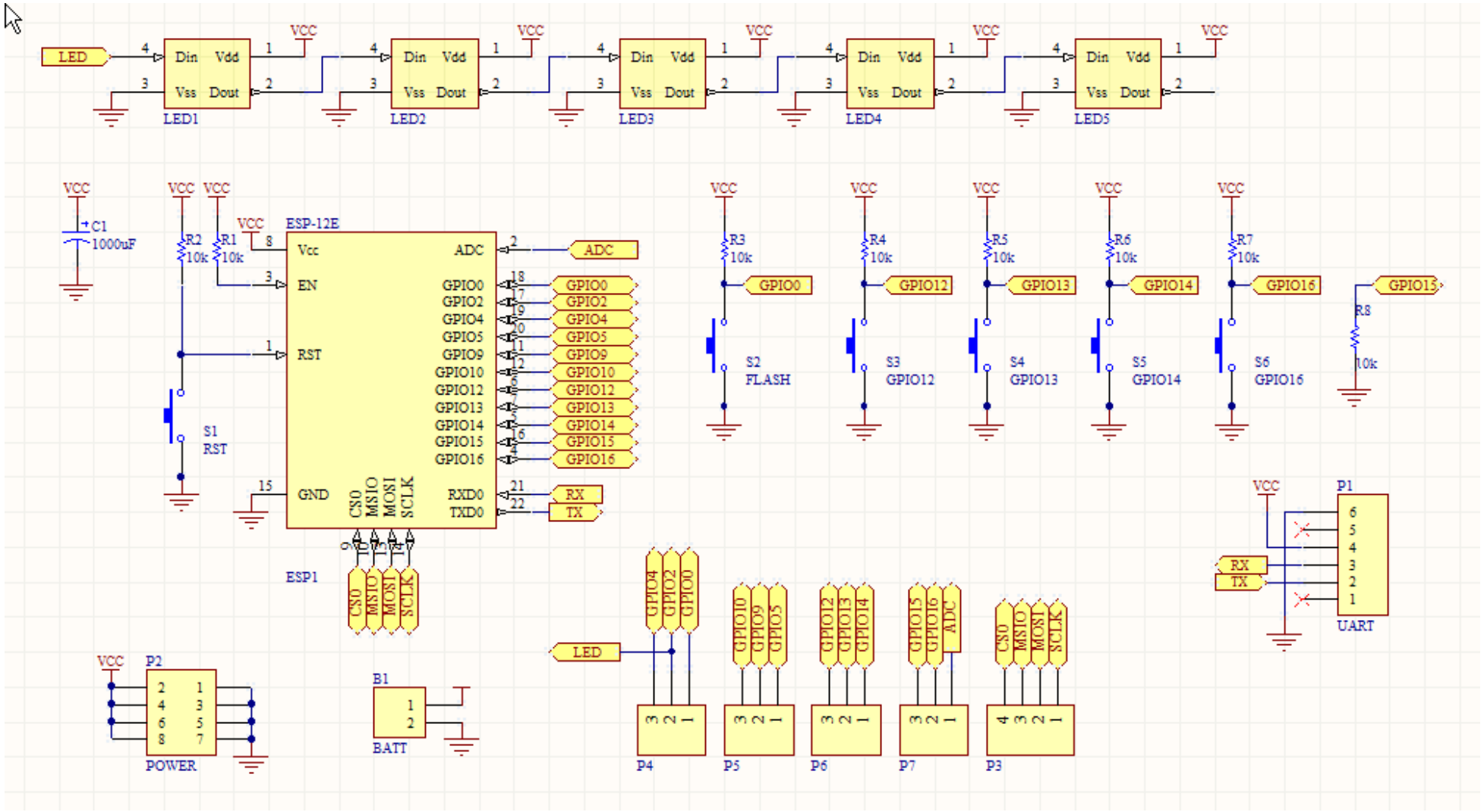
First you'll need to ascertain what device node the adapter came up on (Chances are it'll be /dev/ttyUSB0), do a `dmesg | tail` to confirm.

Download and extract the firmware archive.

```
/path/to/extracted/badge/firmware $ chmod +x *.sh
```

```
/path/to/extracted/badge/firmware $ ./go.sh /dev/<your adapter>
```

And follow the prompts.
There are two separate scripts which are called by go.sh, one “flash.sh” flashes the NodeMCU firmware to the badge, “upload.sh” uploads the badge code to the badge.


LED
Din
Vdd
Vss
Dout
LED1
LED2
LED3
LED4
LED5
VCC
C1
1000uF
R1
R2
10k
ESP-12E
Vcc
EN
RST
GND
ADC
GPIO0
GPIO2
GPIO4
GPIO5
GPIO9
GPIO10
GPIO12
GPIO13
GPIO14
GPIO15
GPIO16
RXD0
TXD0
RX
TX
CS0
MSIO
MOSI
SCLK
ESP1
S1
RST
R3
R4
R5
R6
R7
R8
S2
FLASH
S3
S4
S5
S6
P1
UART
P2
POWER
B1
BATT
P3
P4
P5
P6
P7