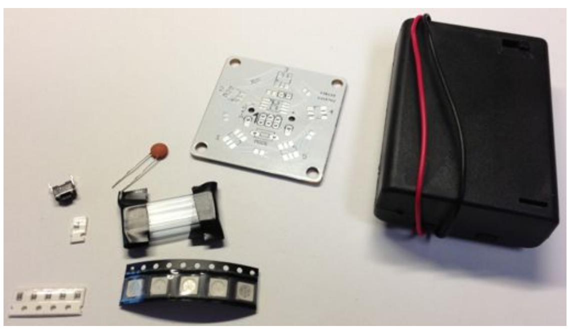
Tiny RGB SMT Build Instructions

PARTS LIST:

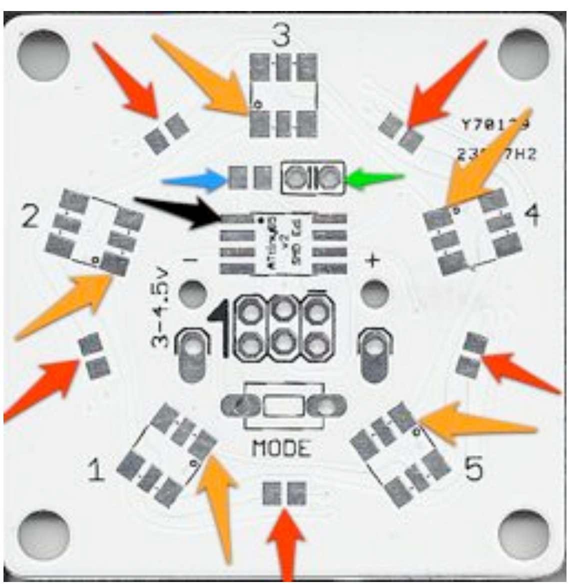
Quantity	Name	Location	Description
1	Circuit Board	-	
5	5050 RGB LEDs	LED1-LED5	
5	68 Ω resistors	-	680
1	10k Ω resistor	-	103
1	100 nF capacitor	-	104
1	ATtiny85	ATTINY	ATTINY85-20SU
1	Tact Switch	MODE	6 x 3 x 4.3 mm
1	3 x AA Battery Box		

BUILD INSTRUCTIONS:

Since this is a slightly advanced build, I'm not going to walk through each step of the process as much as I do in my beginner kits. The instructions for this kit are intended to help guide the build order and any tips I've discovered while building the prototype boards.



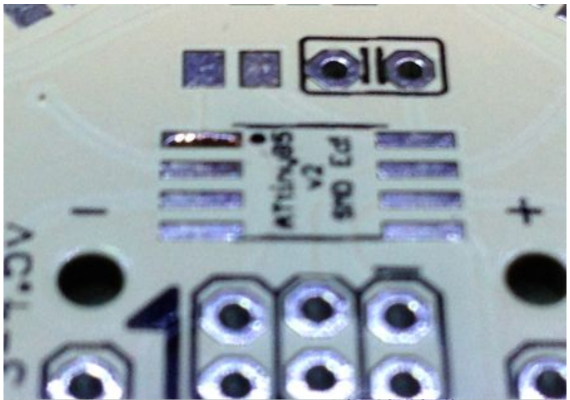
1.The contents of the kit - the circuit board, five 68 ohm resistors on a strip, a solitary 10k resistor, the 100 nF capacitor, the microcontroller in a segment of packing tube and the battery box. Everything other than the PCB is inside the battery compartment, so open it up and carefully extract all the parts.

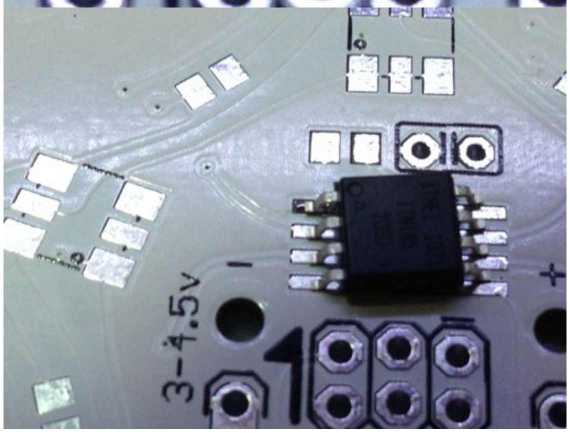


2. A quick overview of the board as it doesn't have much in the way of part labeling on the front:

• The red arrows point to the 68 Ω resistor pads. The 68 Ω resistors are marked "680" on their faces.
• The blue arrow points to the 10k Ω resistor pad. The 10k Ω resistor is marked "103".
• The green arrow points towards the solitary capacitor pad. If you have a 100nF SMD capacitor, you could use the holes for SMD pads, otherwise use the through hole capacitor included with the kit.
• The black arrow points to the microcontroller's first pin. There's a small black dot on the board to indicate the orientation of the microcontroller. If you solder the microcontroller onto the board backwards, it won't work, and it's a pain to get off the board. Always double check the orientation of polarized parts.
• The orange arrows point to pin 1 of the 5050 RGB LEDs.

The 68 Ω resistors act as current limiters for the LEDs. If you swap them for ones with smaller values, the first thing that tends to happen is the microcontroller goes into a constant state of reset. If you let it happen for long enough, you could also burn out the microcontroller or possibly damage an LED. As this is voltage sensitive, if you intend on using this from a 3v source, you can get away with smaller resistors - 27 Ω is comfortable for 3v for example. 68 Ω is plenty for 4.5v. The 10k resistor pulls up the microcontroller's reset pin and helps prevent erratic behavior. The capacitor helps filter power coming from the batteries, but I haven't ever had problems running from batteries without a filtering capacitor, so if you wish to leave it off, the circuit still works without it.





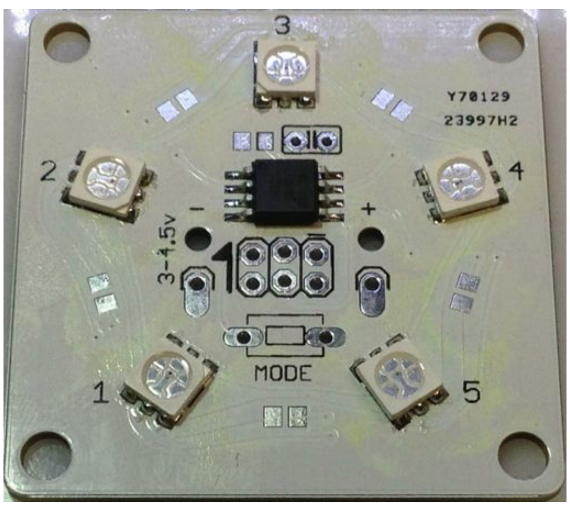
3. Open the battery box and remove the parts. Be careful to not lose the 10k resistor since it's so small. I always start with the microcontroller, since the microcontroller's pads are large enough to get me prepared for surface mount parts. Add a little bit of solder to the first pin on the board and use that to reflow the chip into place. If you need a guide to reflow soldering, I can't recommend Dave Jones' guide highly enough:

http://youtu.be/b9FC9fAlfQE

And Jeremy's guide is also helpful:

http://thecustomgeek.com/2012/06/14/smd-soldering-you-can-do-it/

Either way it boils down to adding a bit of solder to a pin on the board and using that to reflow solder onto the component. The biggest problems are solder control and flux management. You want the finest solder you can find - when I was trying it with 0.032" solder, even the lightest dab of solder added far too much. It's much easier to control how much solder you add with finer solders - I ended up with 0.015" silver solder because the 0.032" was too hard to manage, and it worked out fine. Solder all 8 legs of the microcontroller before moving on to the next step. Check it for shorts and poorly soldered joints with magnification. I do all of my work with a head visor that doesn't really provide that much magnification - I think it's only good for about 3x, but it's enough to see the components up close.

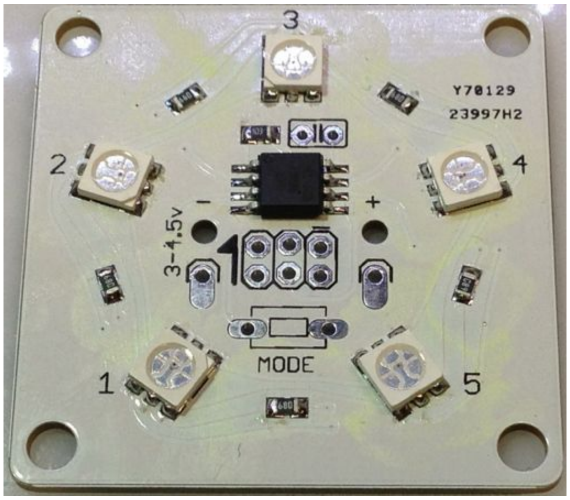


4. Add flux to the board and start soldering the LEDs into place. The LEDs have a little notch in them to indicate pin 1 as shown in this picture:

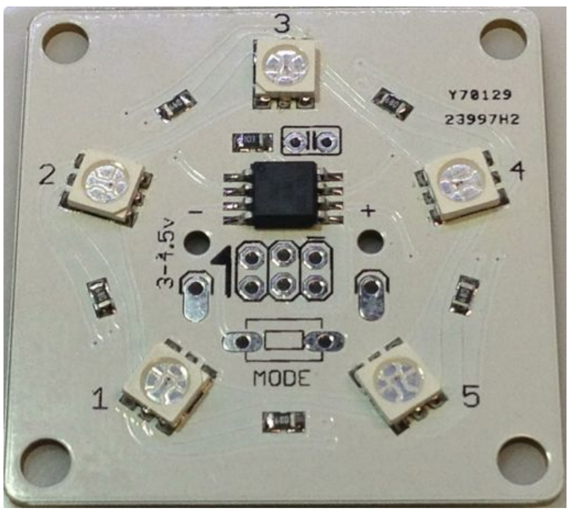
The board indicates pin 1 with a little black dot. Make absolutely certain that the LED is in the right orientation before you start to solder it into place - the leads on the LEDs make removing them a difficult proposition once they're all soldered into place.

I found it easiest to solder them into place by starting with pins 2 and 5, the two skinnier pads in the middle. That way, if the LED ends up slightly off center, the opposite pad won't have enough available pad to start to solder to, and it's easier to reheat the opposite pad to assist in moving the LED. Flood pad 2, tack the LED into place, then snap it into place using your finger and the iron. The clear part of the LED scratches very easily - I never felt comfortable pressing it into place with the tweezers. Make sure you have the LED centered and rotated properly before proceeding to pin 5. If you can't get pin 5 to flow, try to move the LED a little bit by reflowing pin 2. The LED can take a surprising amount of heat, but be careful just the same. I haven't cooked off any, but the bond wire is very fine and it's connected directly to the lead so it wouldn't take much. Once pins 2 and 5 are soldered, proceed to pins 1, 3, 4 and 6. Once you have one LED on the board, you can do the other 4 the same way.

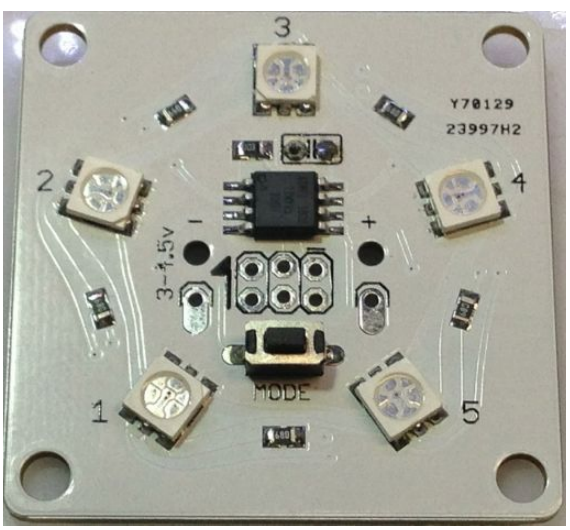
I did the work with a reasonably broad chisel tip as well as a fine tip and honestly the fine tip wasn't any easier than the broad tip. The fine point tip was absolutely useless on the through hole components, though.



5. Solder on the resistors. After the LEDs, they're pretty simple. Flood one pad, tack it into place, then press it down onto the board before soldering the other pad.

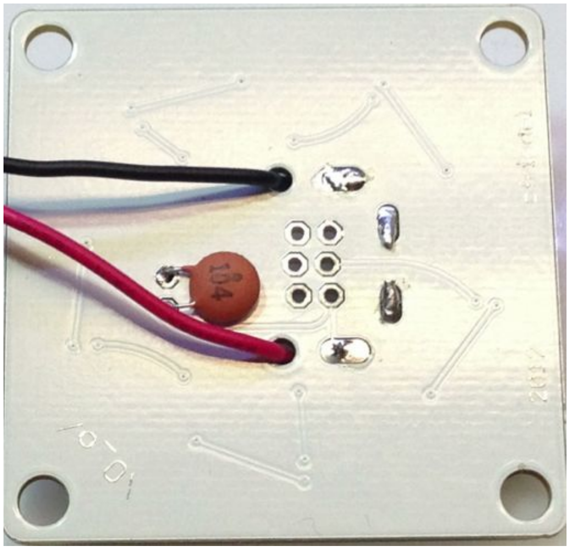


6. Clean the accumulated flux off of your board before soldering on the switch. Use whatever your flux requires - mine cleans up nicely with rubbing alcohol, yours might need something different.

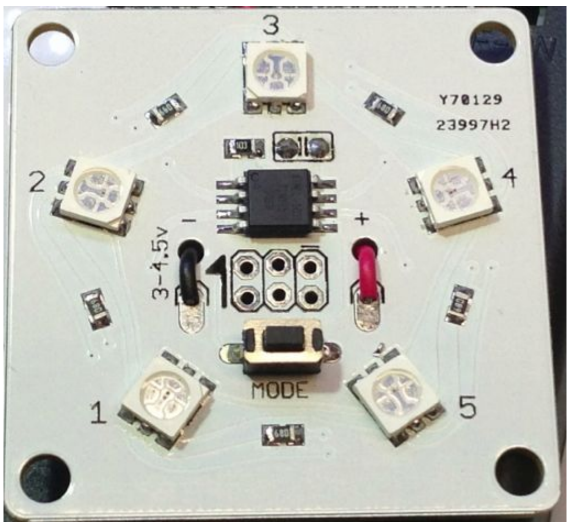


7. Solder the Mode switch into position. It isn't polarized, so it doesn't matter which way you solder it into place.

At this point, the board is done. If you want to power it from a USB cable, you should be able to power it directly to the negative and positive terminals on the board. If you want to power it from the included battery box, follow along on the next page.



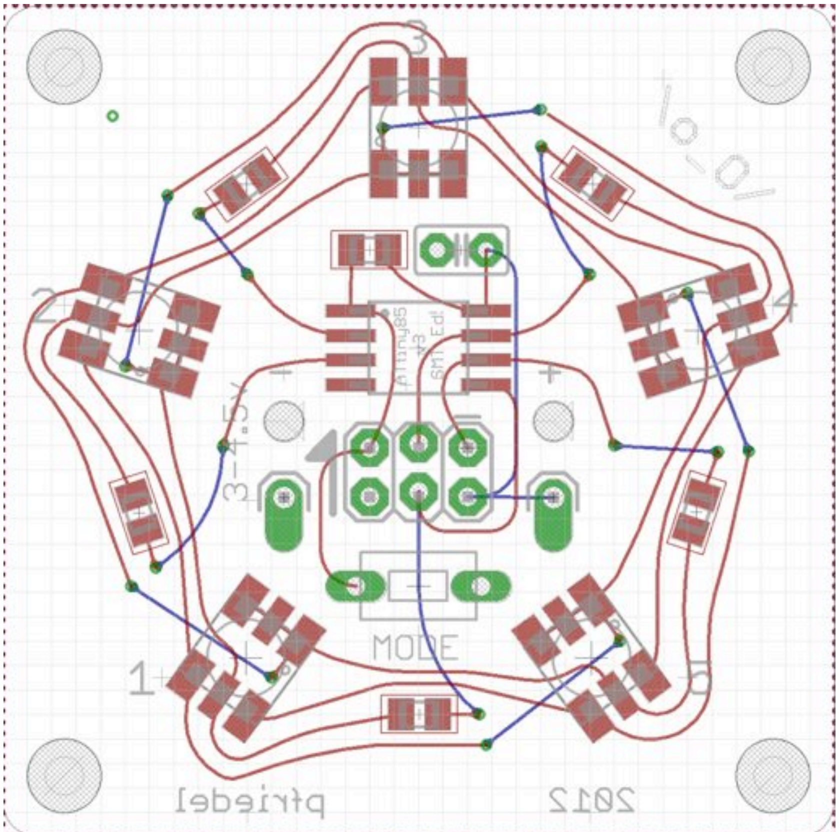
8. Thread the black negative lead up from the back of the board to the front of the board and do the same for the red positive lead. Adjust the length of the wires so there isn’t too much extra loose wire. At this point I usually try to make a mark on the wire where it passes over the hole and where it passes over the lower bound of the solder pad. Then I unthread the wire and trim it to length with a wire cutter and strip back the insulation and tin the exposed wire.



9.Once you have trimmed and tinned the battery wires, thread them up from the back of the board again and loop them over to the solder pads as shown on the left.

Attach the board to the battery box with double sided sticky tape or hot glue. The board is now finished. Insert batteries and flip the power switch and it should start from mode 1.

TROUBLESHOOTING:



The resistors aren't labeled on the board, but they're all counter-clockwise from their associated LED. So R1 is below the Mode switch, R2 is clockwise from that, R3 is clockwise from that, etc. To test LED function, 4.5v through more than 130 Ω resistance should light up the LEDs listed in the table to the right. Unlisted connections might produce some light (sometimes across 2 LEDs, even), but those aren't ever called out in the code, so don't worry about them if they light up.

The LEDs are arranged such that pins 1 and 6 are blue, pins 2 and 5 are red and 3 and 4 are green.

Negative	Positive	LED / Color
1	2	3 / Green
1	3	4 / Blue
1	5	1 / Red
2	3	4 / Green
2	4	5 / Blue
2	1	2 / Red
3	4	5 / Green
3	5	1 / Blue
3	2	3 / Red
4	5	1 / Green
4	1	2 / Blue
4	3	4 / Red
5	1	2 / Green
5	2	3 / Blue
5	4	5 / Red

DISPLAY MODES:

Each time you hit the MODE button, the board will advance modes. The board will tell you which mode you are entering by lighting up the LED associated with that mode which is next to a number on the board. 1-5 are indicated by red LEDs while 6-10 are indicated by green LEDs.

1. A slow walk through all the hues of the color spectrum.
2. A faster walk through the color hues.
3. Even faster than #2.
4. A slow progression with every color mapped one to one to LEDs. While modes 1-3 work with a wide virtual LED space, this mode and the next one work only with the physical LEDs.
5. A faster version of #4.
6. Random colors at random positions, changing every second or so.
7. Bouncing up and down in hue saturations, slowly walking through hues.
8. The same as #7, with a faster hue progression.
9. Displays the red, green and blue LEDs directly without mixing. Useful for spotting a bad LED.
10. Randomly selects any of the prior 9 modes and displays it for 5 minutes before selecting the next mode.

The system will automatically enter a low power sleep mode after 4 hours of modes 1-9, but mode 10 will run until the batteries run down. If you hit the MODE button, the system will wake up and resume the running the previous mode for another 4 hours. If you hit the MODE button again while it is still indicating which mode it is about to enter, it will advance to the next mode.

The low power mode is drawing just under 2 μA, so the power switch isn't strictly necessary - 2 μA is on the order of the battery's self discharge rate.

In normal use the board draws about 20 mA - the expected lifespan is about a week nonstop - 168 hours. As the batteries die, the LEDs will slowly start to fade and each color will start to fail one by one. If you only use the 4 hour timed mode once a day, each set of batteries will last over a month given good alkaline cells.

SOURCE CODE:

The software repository for this project is at https://github.com/pfriedel/TinyFiveCircle

Any questions? Mail me - pfriedel@compulsive.net.