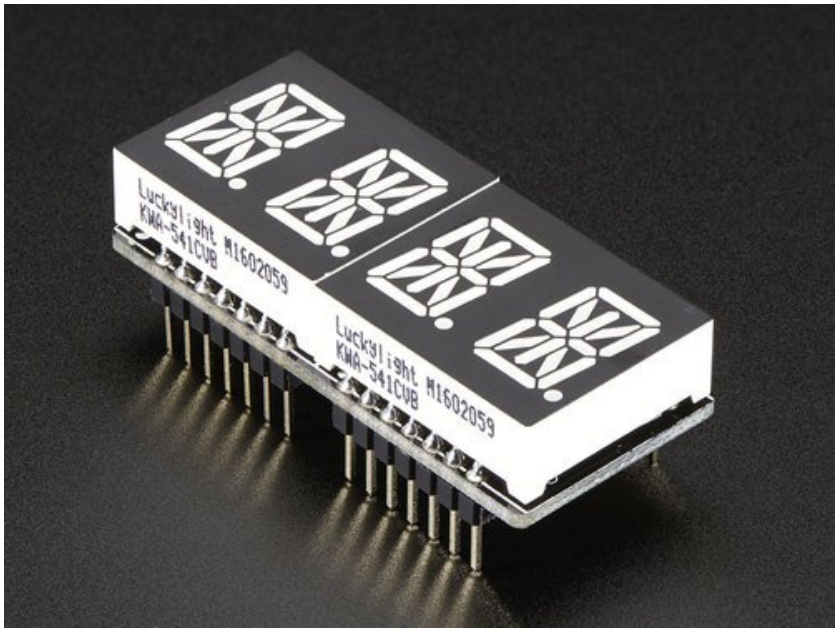
adafruit learning system

14-Segment Alpha-numeric LED FeatherWing

Created by lady ada



Last updated on 2017-07-14 05:22:47 AM UTC

Guide Contents

Guide Contents 2
Overview 3
Pinouts 7
Power Pins 8
I2C pins 8
Address Jumpers 9
- Changing Addresses 10

Assembly 11
- Prepare the header strips: 11
- Add the FeatherWing PCB: 12
- And Solder! 13

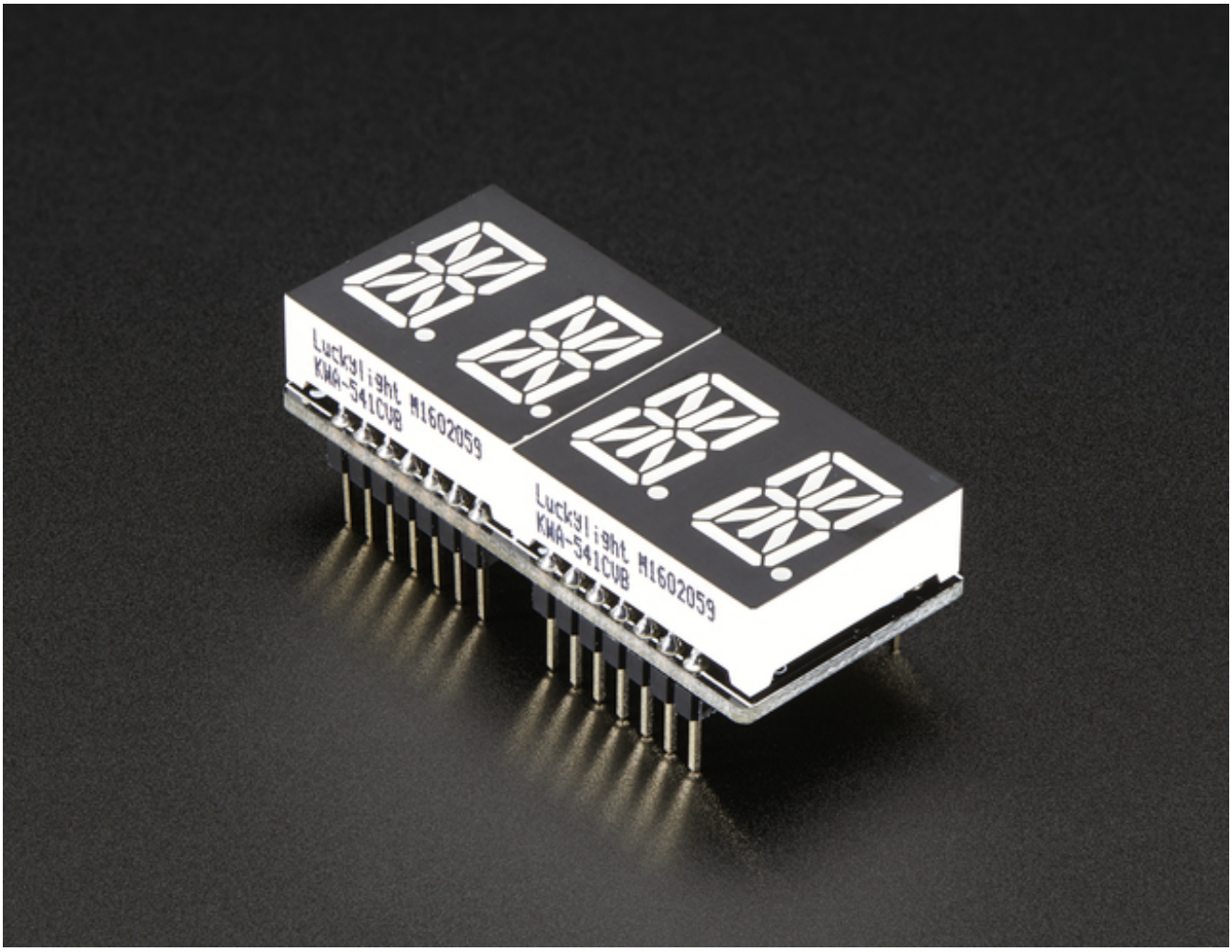
Usage 22
Install Adafruit GFX 23
Run Test! 23
Library Reference 26
ASCII data 28
Writing Data 28
Downloads 29
Schematic 29
Fabrication Print 29

Overview

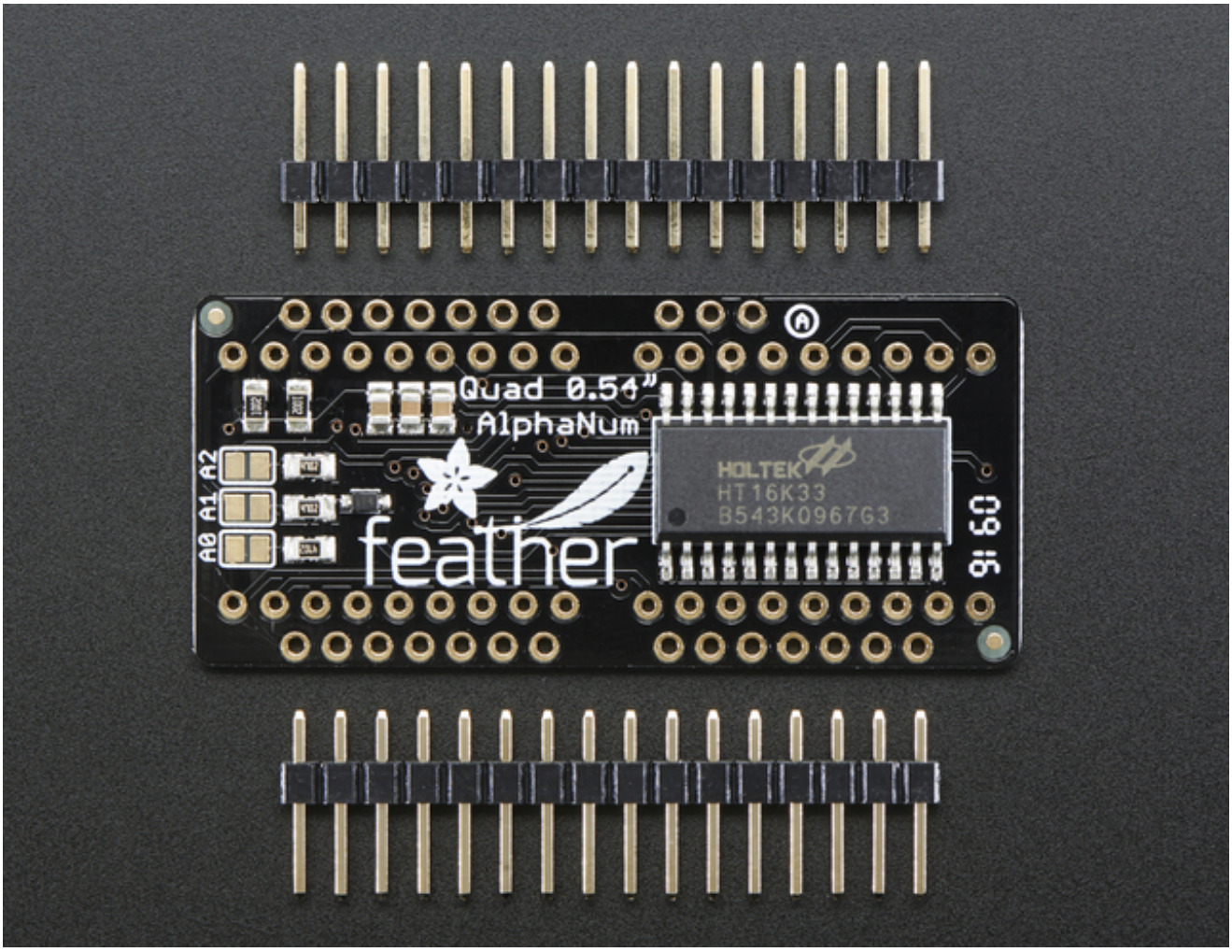
Display, elegantly, 012345678 or 9! Gaze, hypnotized, at ABCDEFGHIJKLM - well it can display the whole alphabet. You get the point.

This is the **Adafruit 0.56" 4-Digit 14-Segment Display w/ FeatherWing Combo Pack**. Available in Blue, Green, Red, White, and Yellow. It is a nice, bright alphanumeric display that shows letters and numbers in a beautiful hue. It's super bright and designed for viewing from distances up to 23 feet (7 meters) away. Each of the digit sets have 14 segments on a dark background and we give you a set of two alphanumeric displays as well as a Featherwing driver board so you can make a clock or a four letter word.

Works with any and all Feathers!



14-Segment Matrices like these are 'multiplexed' - so to control all the seven-segment LEDs you need 18 pins. That's a lot of pins, and there are driver chips like the MAX7219 (http://adafru.it/453) that can control a matrix for you but there's a lot of wiring to set up and they take up a ton of space. Wouldn't it be awesome if you could control a matrix without tons of wiring? That's where these Alphanumeric LED Matrix FeatherWings come in, they make it really easy to add a 4-digit alphanumeric display with decimal points.

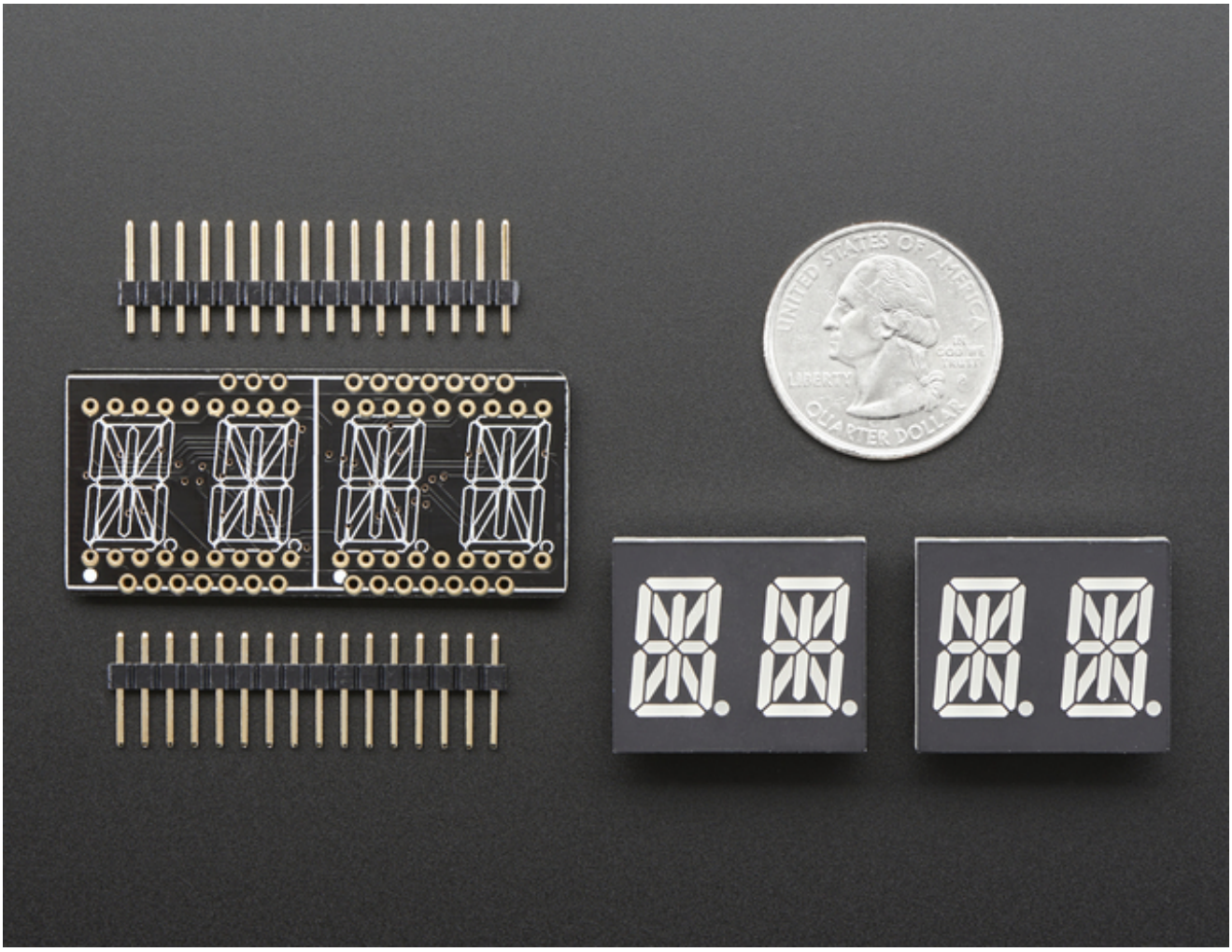


The LEDs themselves do not connect to the Feather. Instead, a matrix driver chip (HT16K33) does the multiplexing for you. The Feather simply sends i2c commands to the chip to tell it what LEDs to light up and it is handled for you. This takes a lot of the work and pin-requirements off the Feather. Since it uses only I2C for control, it works with any Feather and can share the I2C pins for other sensors or displays.

The product kit comes with:

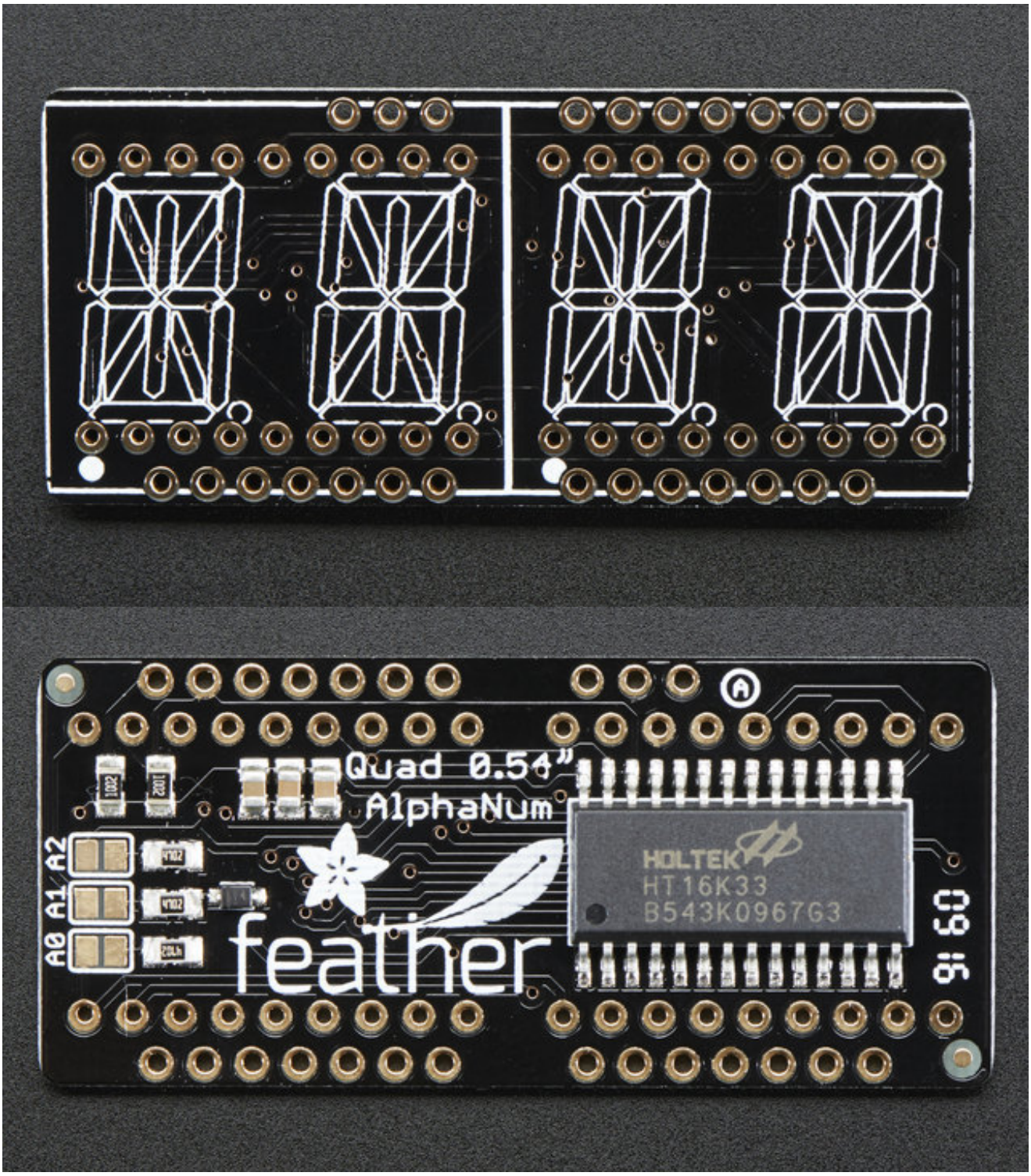
- A fully tested and assembled Adafruit 4-Digit 14-Segment LED Matrix Display FeatherWing
- Ultra-bright 4-digit 0.54" tall segmented display
- Two sixteen pin headers

A bit of soldering is required to attach the matrix onto the FeatherWing but its very easy to do and only takes about 5 minutes!



Of course, in classic Adafruit fashion, we also have a detailed tutorial showing you how to solder, wire and control the display. We even wrote a very nice library for the backpacks so you can get running in under half an hour, displaying images on the matrix or numbers on the 7-segment (http://adafru.it/aLI). If you've been eyeing matrix displays but hesitated because of the complexity, his is the solution you've been looking for!

Pinouts

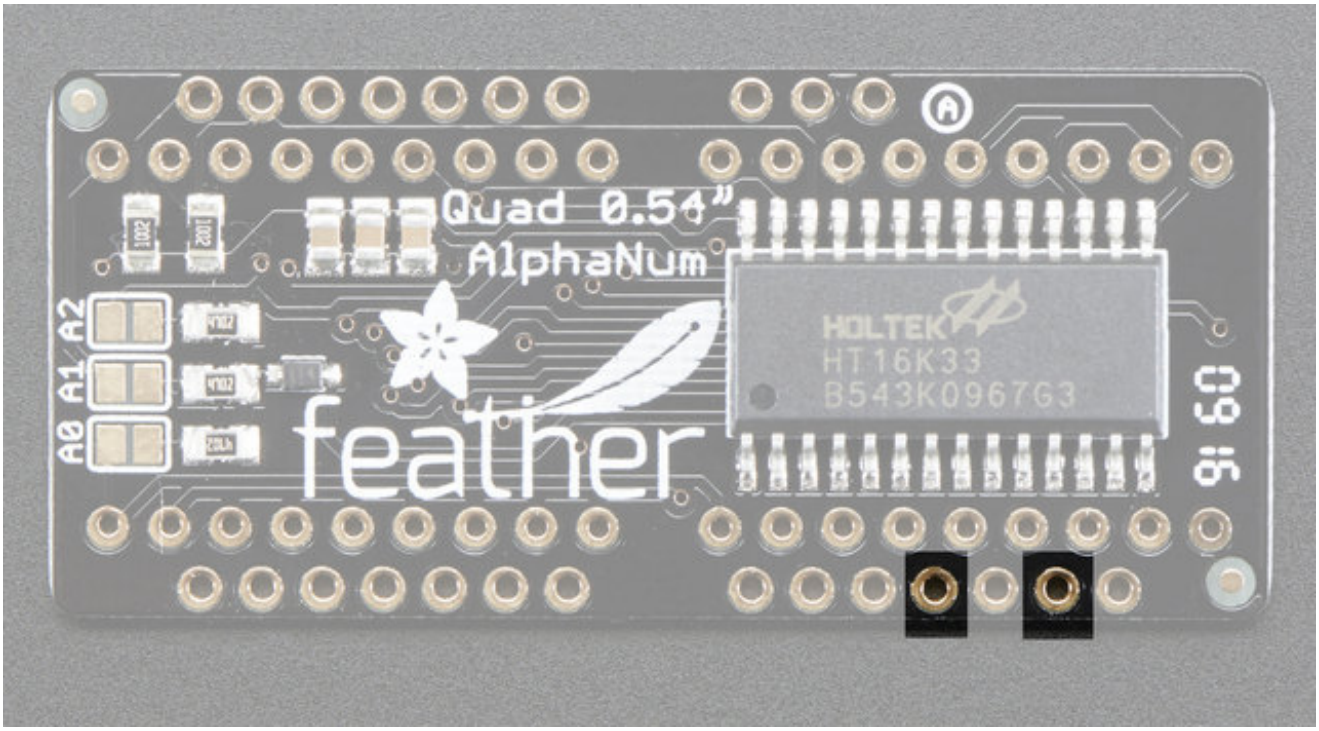


The 14-segment backpack makes it really easy to add a 4-digit numeric display with

decimal points

The LEDs themselves do not connect to the Feather. Instead, a matrix driver chip (HT16K33) does the multiplexing for you. The Feather simply sends i2c commands to the chip to tell it what LEDs to light up and it is handled for you. This takes a lot of the work and pin-requirements off the Feather. Since it uses only I2C for control, it works with any Feather and can share the I2C pins for other sensors or displays.

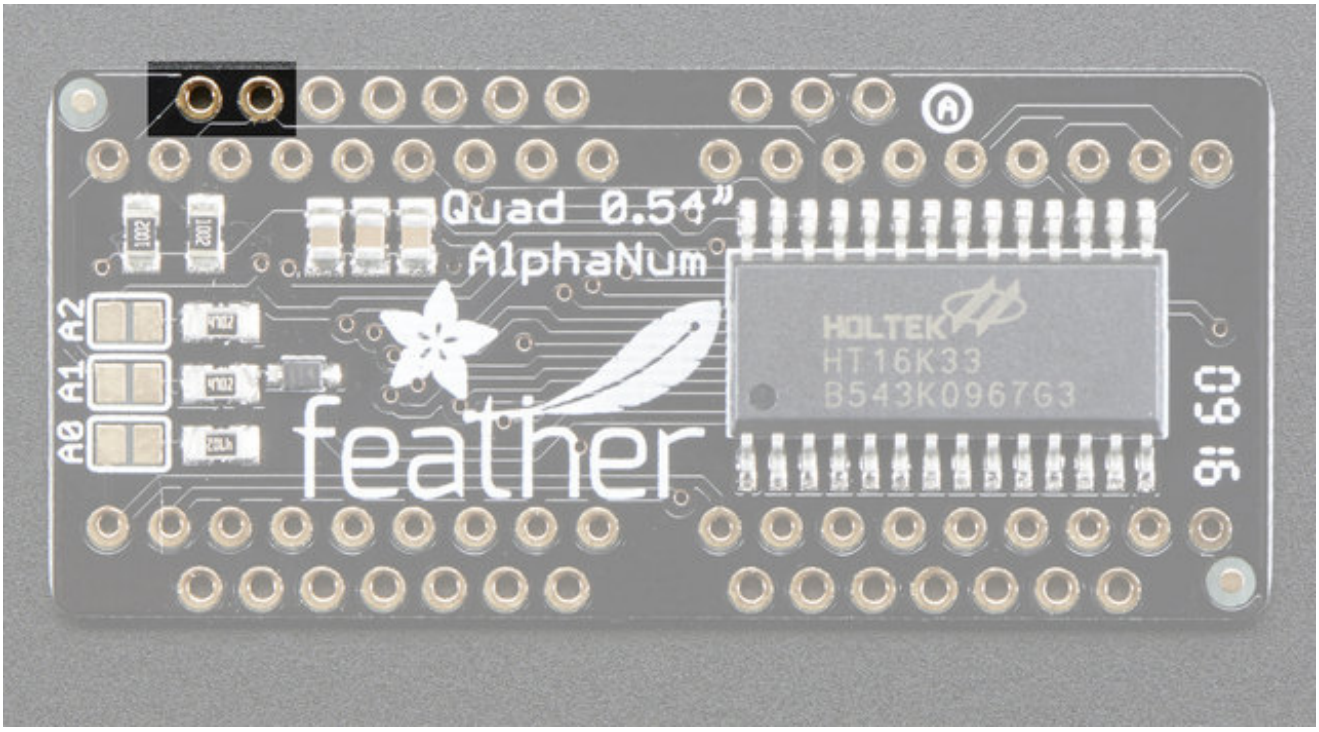
Power Pins



The LED matrix uses only the **3V** (second from the right, above) and **GND** (fourth from the right) pins for power and logic. Current draw depends on how many LEDs are lit but you can approximate it as about 80mA for most uses.

Note that the 3.3V power supply is a tiny bit lower than the forward voltage for the pure green, blue and white LED matrices but we didn't find any significant degredation in brightness. Really, they're still very bright.

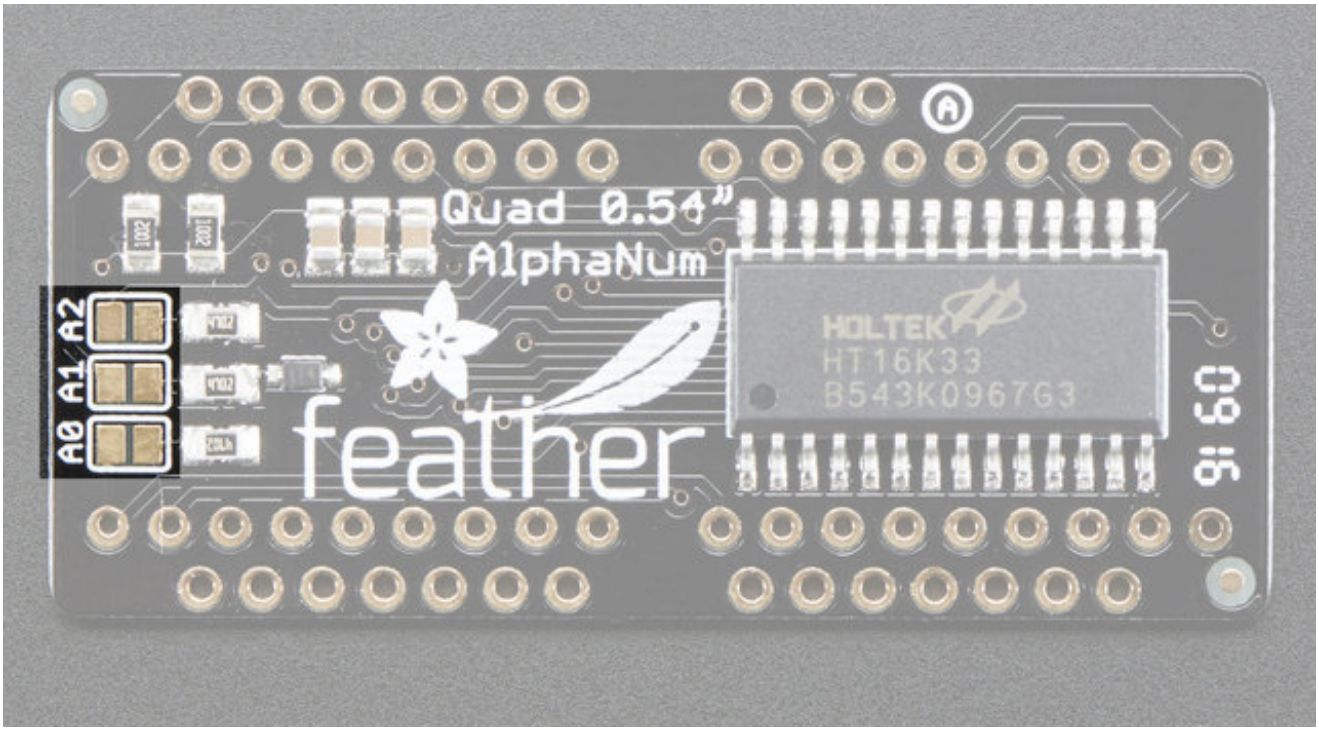
I2C pins



All LED control is done over I2C using the HT16K33 interface library. This means**SDA** (leftmost) and **SCL** (to the right of SDA) must be connected, see above for those pins.

The default address is **0x70** but you can change the address to**0x71-0x77** by bridging solder onto the address pins.

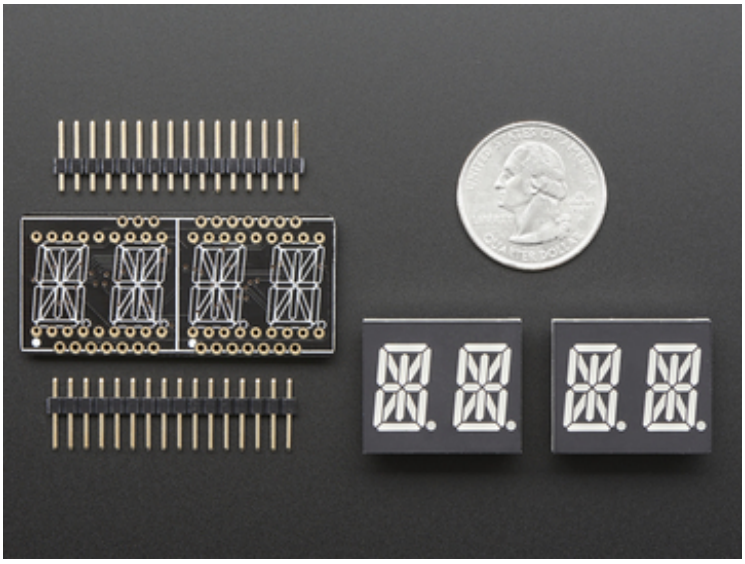
Address Jumpers



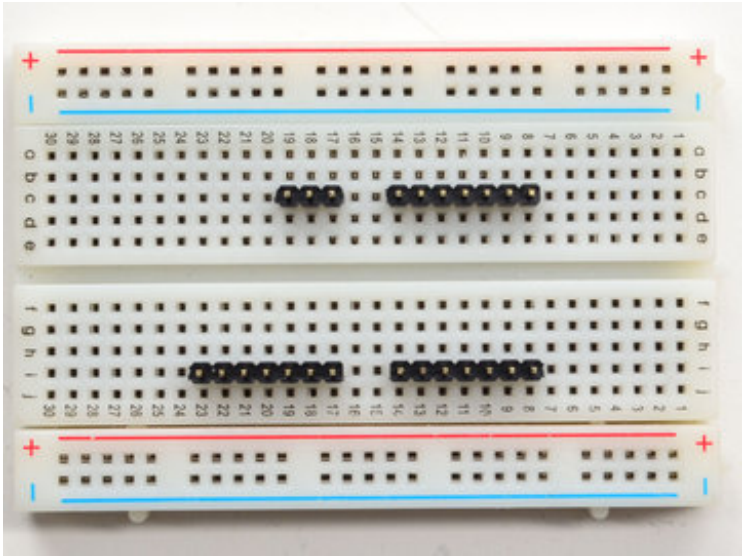
Changing Addresses

You can change the address of a backpack very easily. Look on the back to find the two or three **A0, A1 or A2** solder jumpers. Each one of these is used to hardcode in the address. If a jumper is shorted with solder, that sets the address. **A0** sets the lowest bit with a value of **1**, **A1** sets the middle bit with a value of**2** and **A2** sets the high bit with a value of**4**. The final address is **0x70 + A2 + A1 + A0** So for example if **A2** is shorted and **A0** is shorted, the address is **0x70 + 4 + 1 = 0x75.** If only A1 is shorted, the address is **0x70 + 2 = 0x72**

Assembly

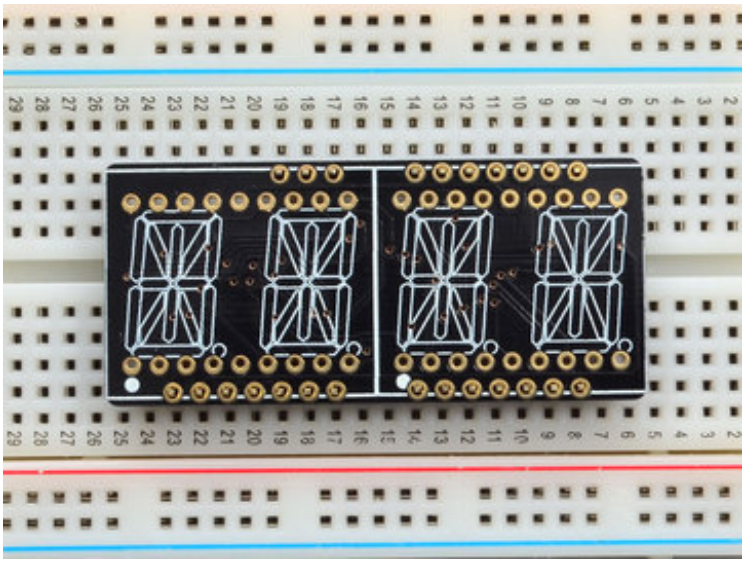
- 

When you buy a pack from Adafruit, it comes with the fully tested and assembled backpack as well as a two 14-segment dualdisplays in one of the colors we provide (say, red, yellow, blue or green). You'll need to solder the matrix onto the backpack but it's an easy task.

- 

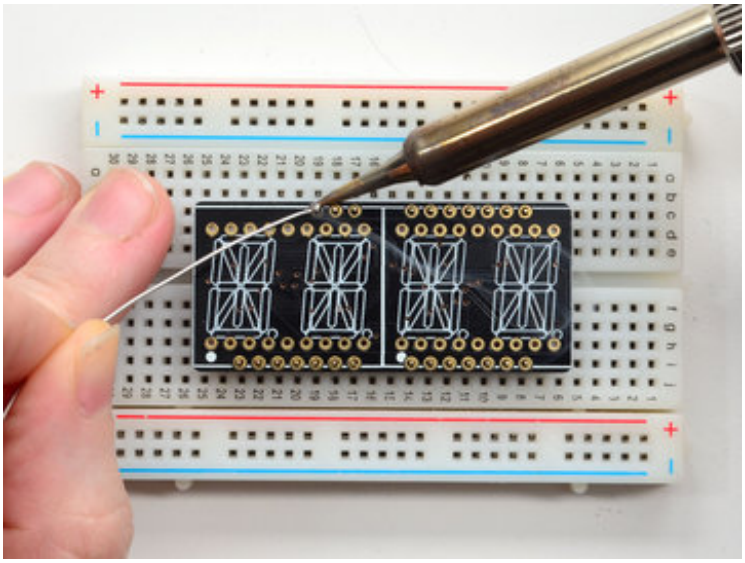
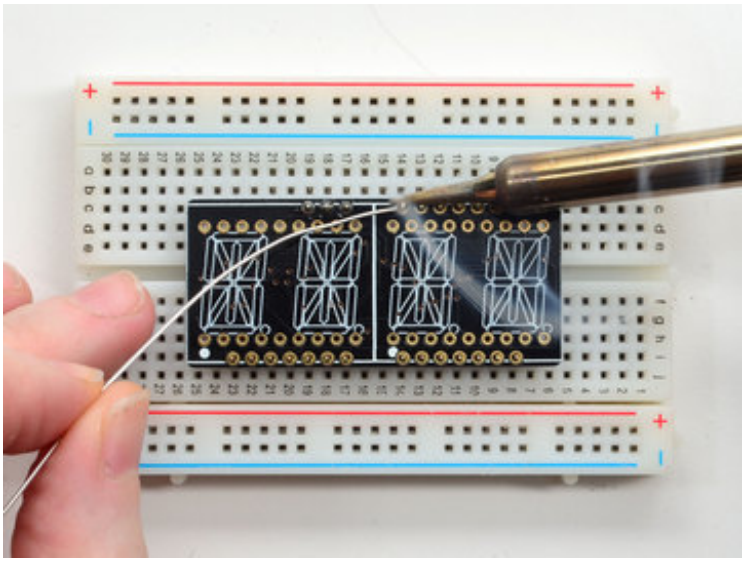
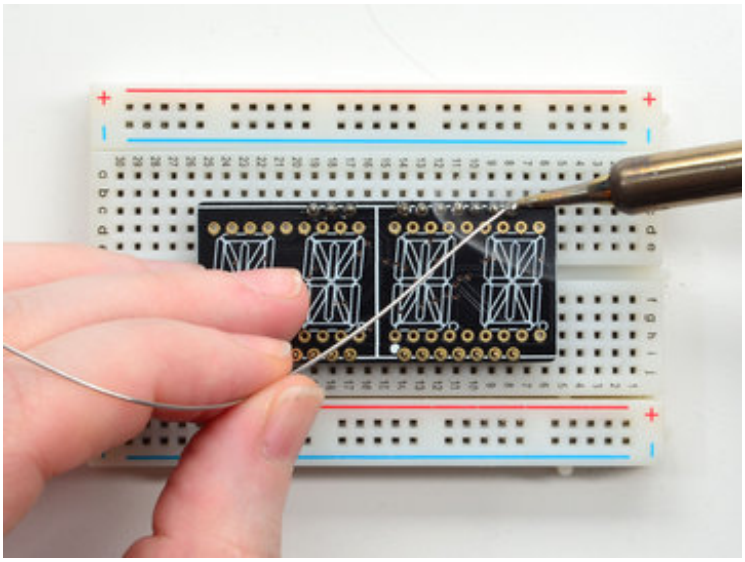
Prepare the header strips:

You'll need three 7-pin and a 3-pin strip of header to attach the Featherwing to your Feather. Cut the header strip to length if necessary. It will be easier to solder if you insert it into a breadboard - **long pins down**

- 

Add the FeatherWing PCB:

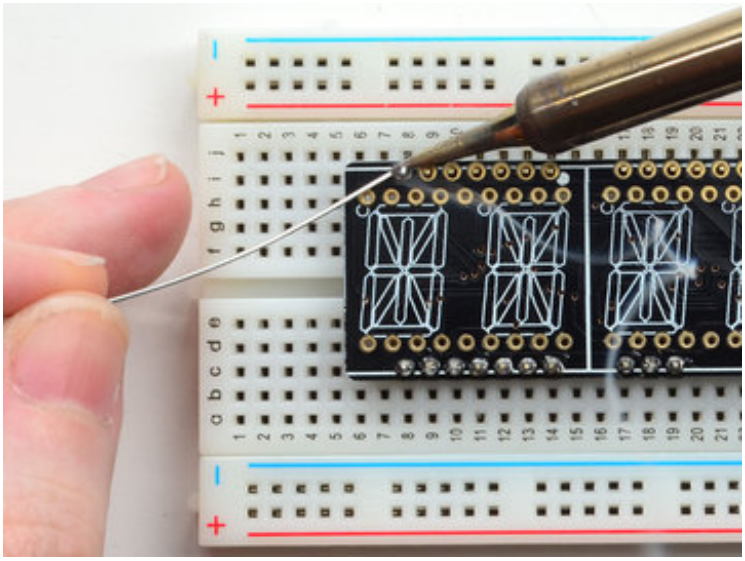
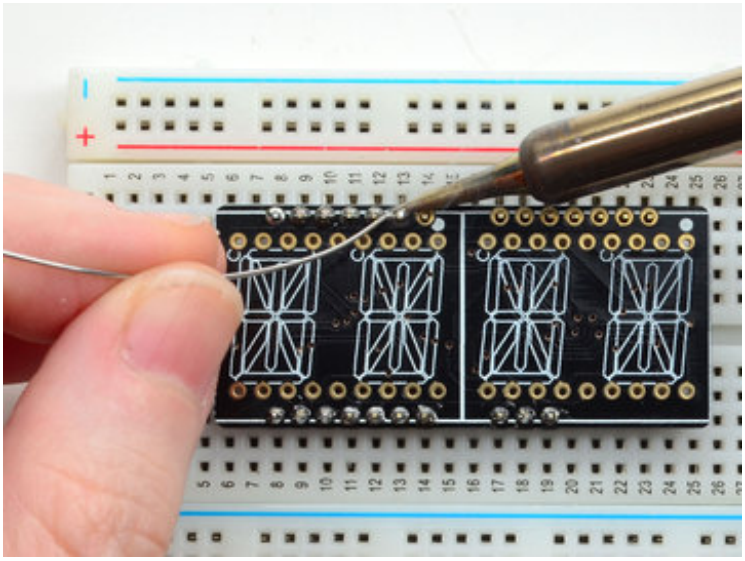
Place the circuit board over the pins so that the short pins poke through the breakout pads

- 
- 
- 

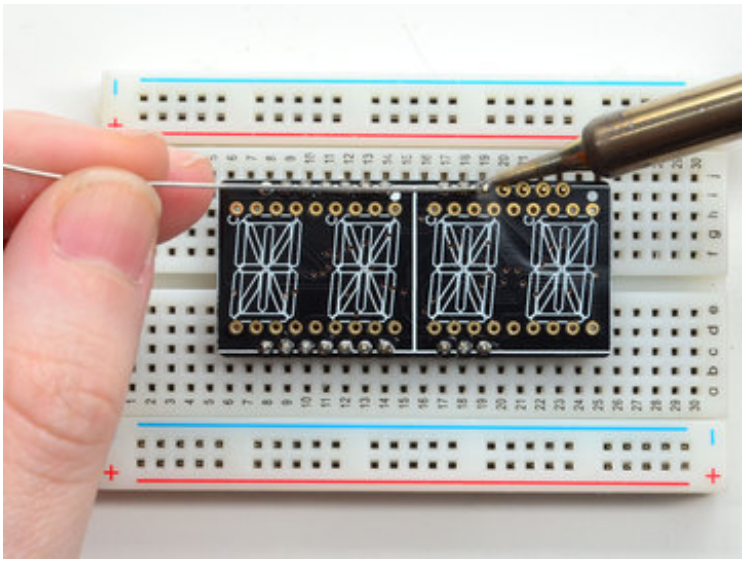
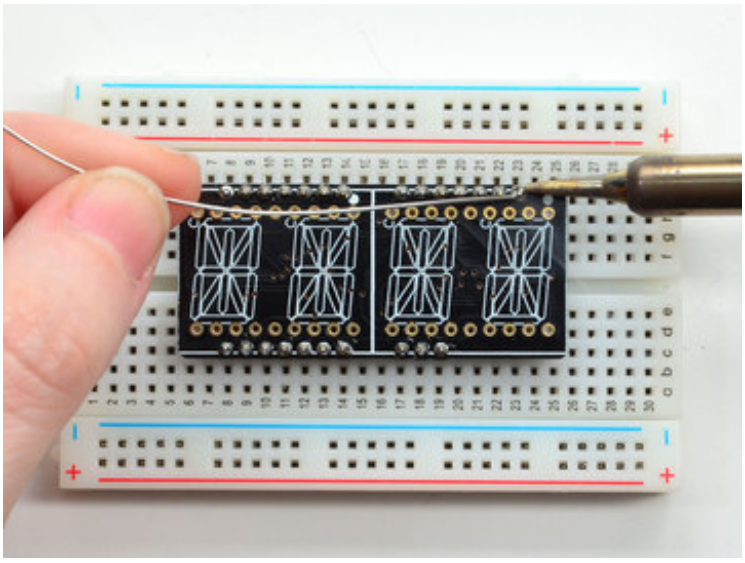
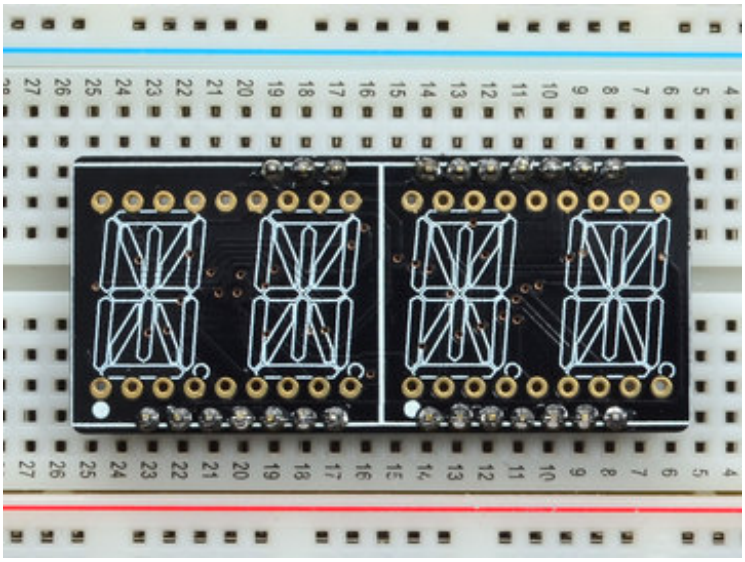
And Solder!

Be sure to solder all pins for reliable electrical contact.

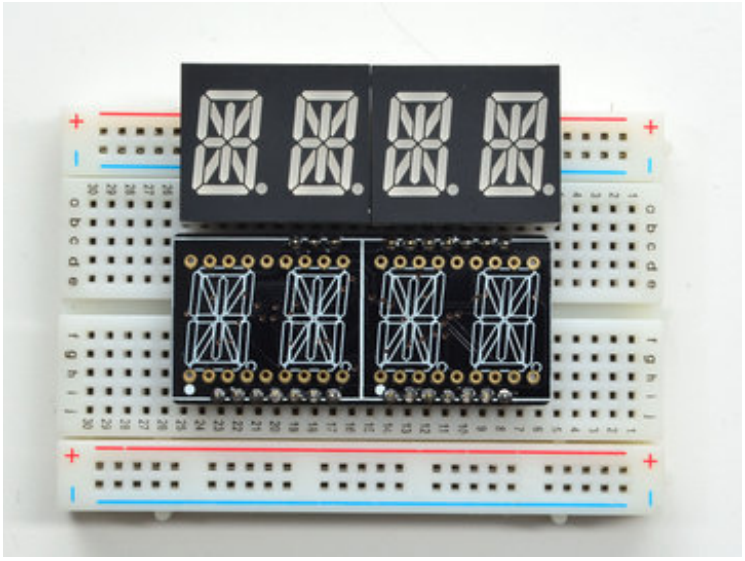
(For tips on soldering, be sure to check out our [Guide to Excellent Soldering](http://adafru.it/aTk)).

- 
- 

Do both all 4 header strips completely!

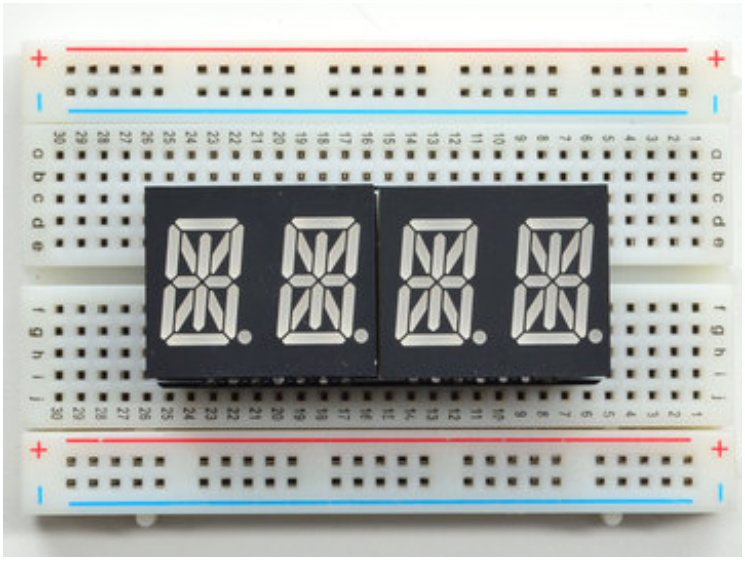
- 
- 
- 

Check your solder joints visually and continue onto the next step

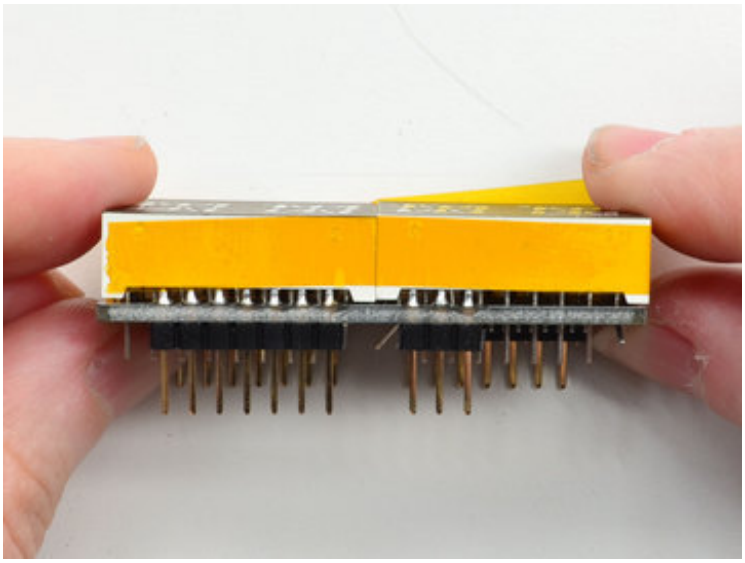
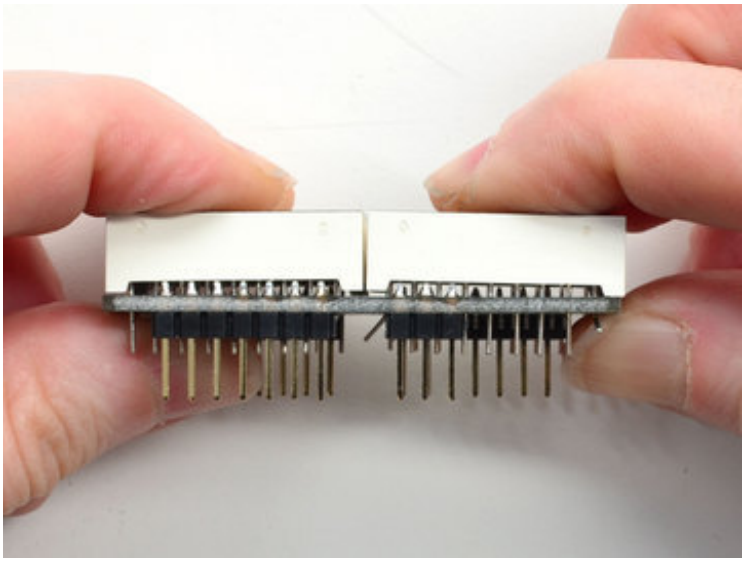
- 

Next we'll attach the LED matrices
Check the Matrices and line it up next to the FeatherWing

Be careful to NOT PUT THE DISPLAYS ON UPSIDE DOWN OR IT WONT WORK!! Check the image to make sure the 'decimal point' dots are on the bottom, matching the silkscreen.

- 

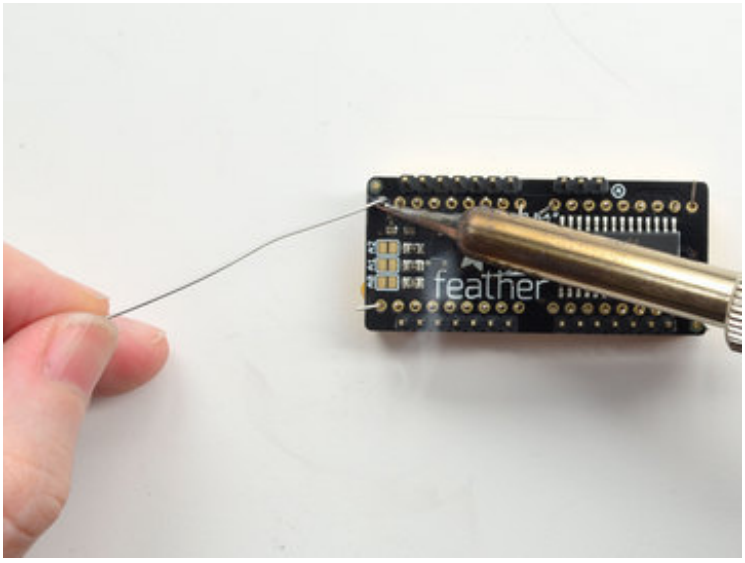
Slot the LED module on top, check again that you have it right way up!

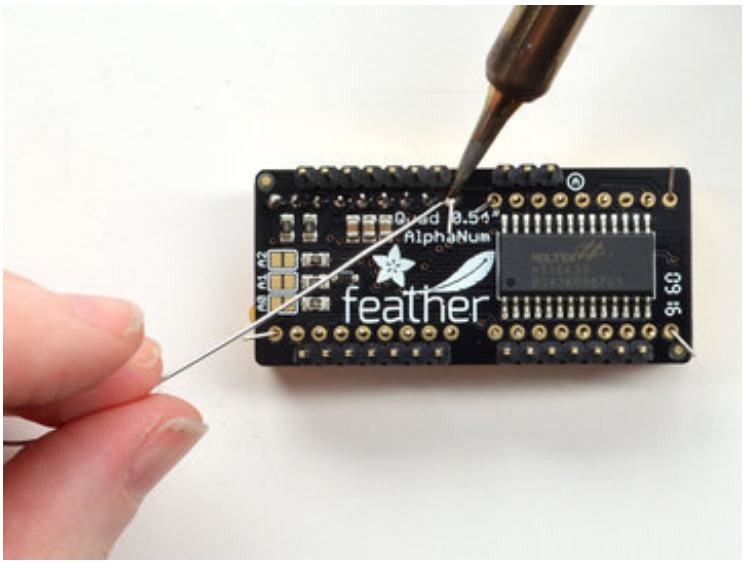
- 
- 

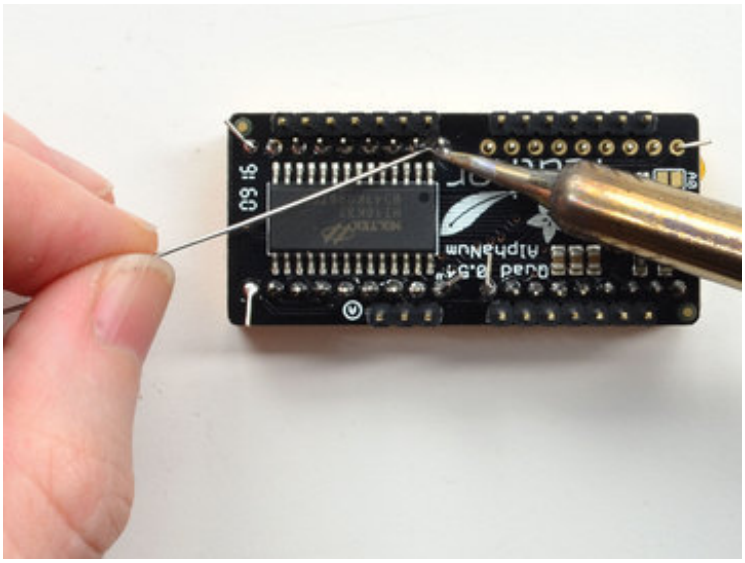
Since there's two modules and they don't sit completely flat on the PCB you may want to use a little tape to bind the modules together to minimize any gap in between.

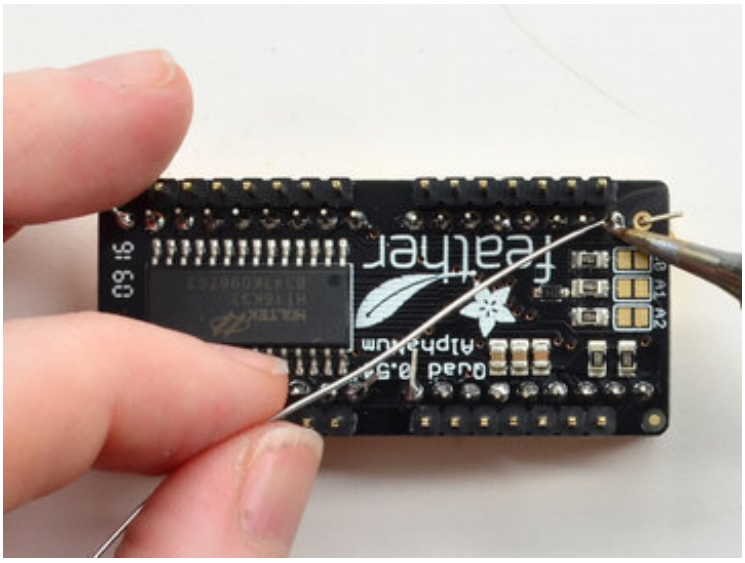
-

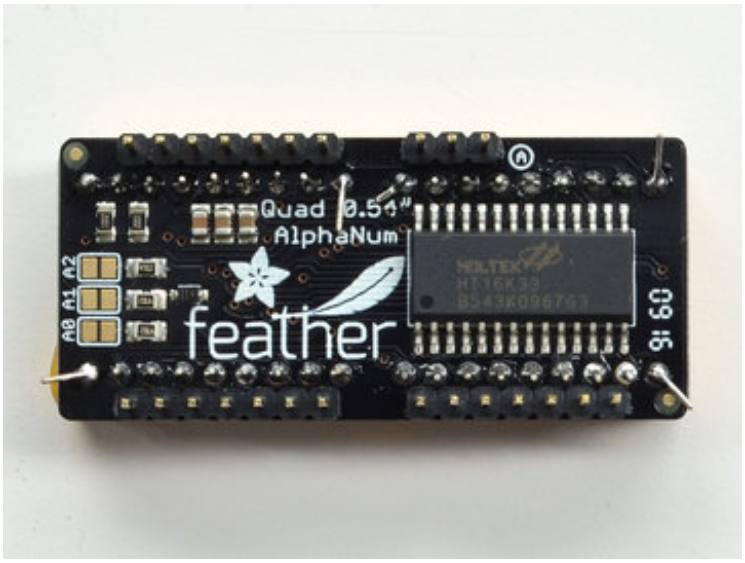
Flip over the kit and bend a few of the LED matrix legs out to keep the LED from slipping out while you solder

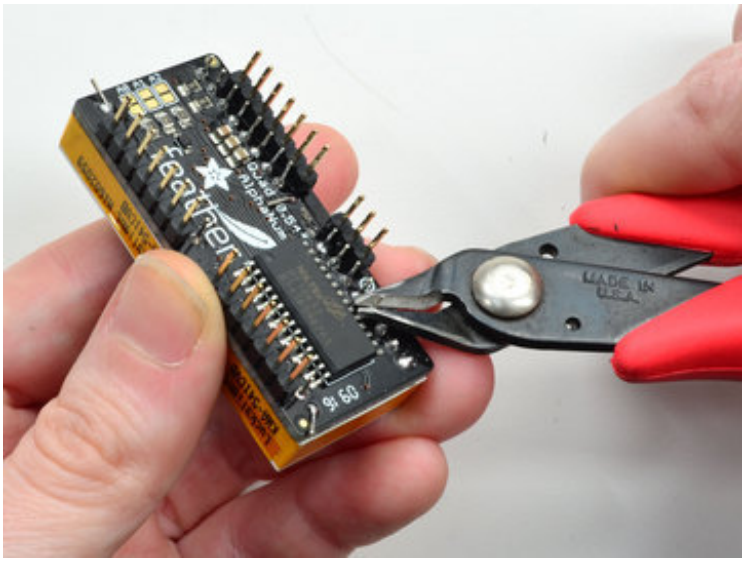
- 

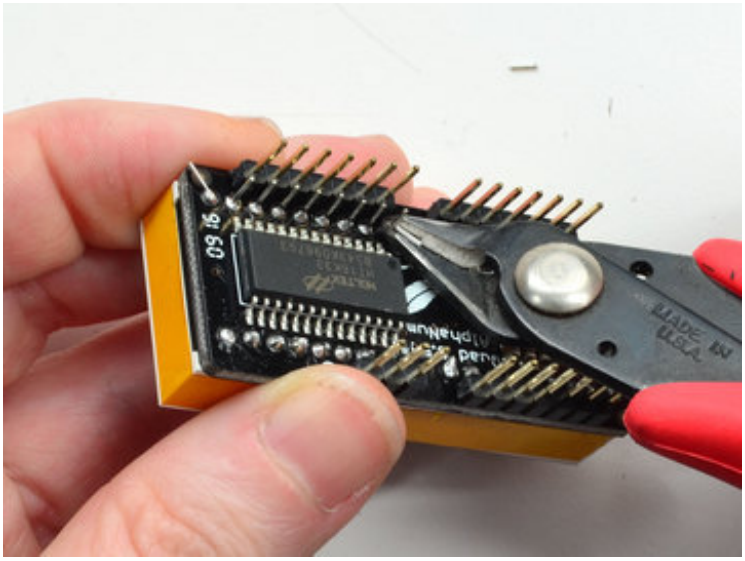
- 

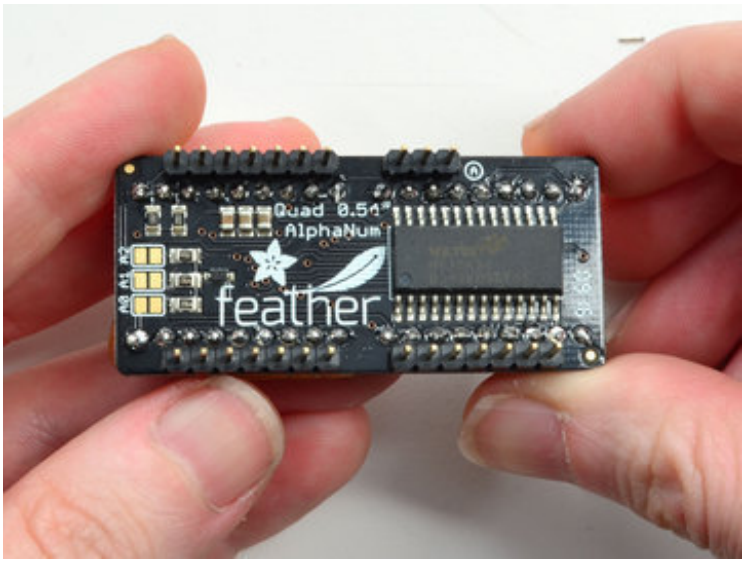
- 

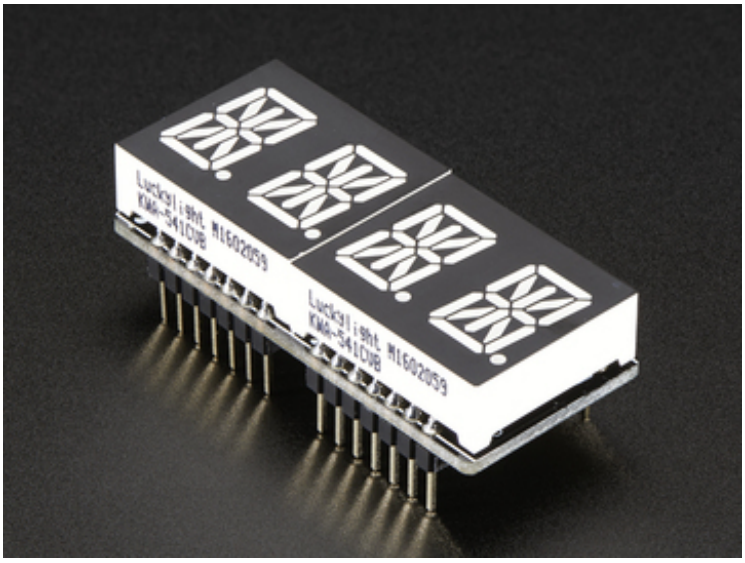
Solder the matrix in place. Note this will be a little tougher because there's not as much space. Go slow, do one pin at a time and you can clip it after each point if you need!
- 

- 

Check your solder joints visually and continue onto the next step
- 

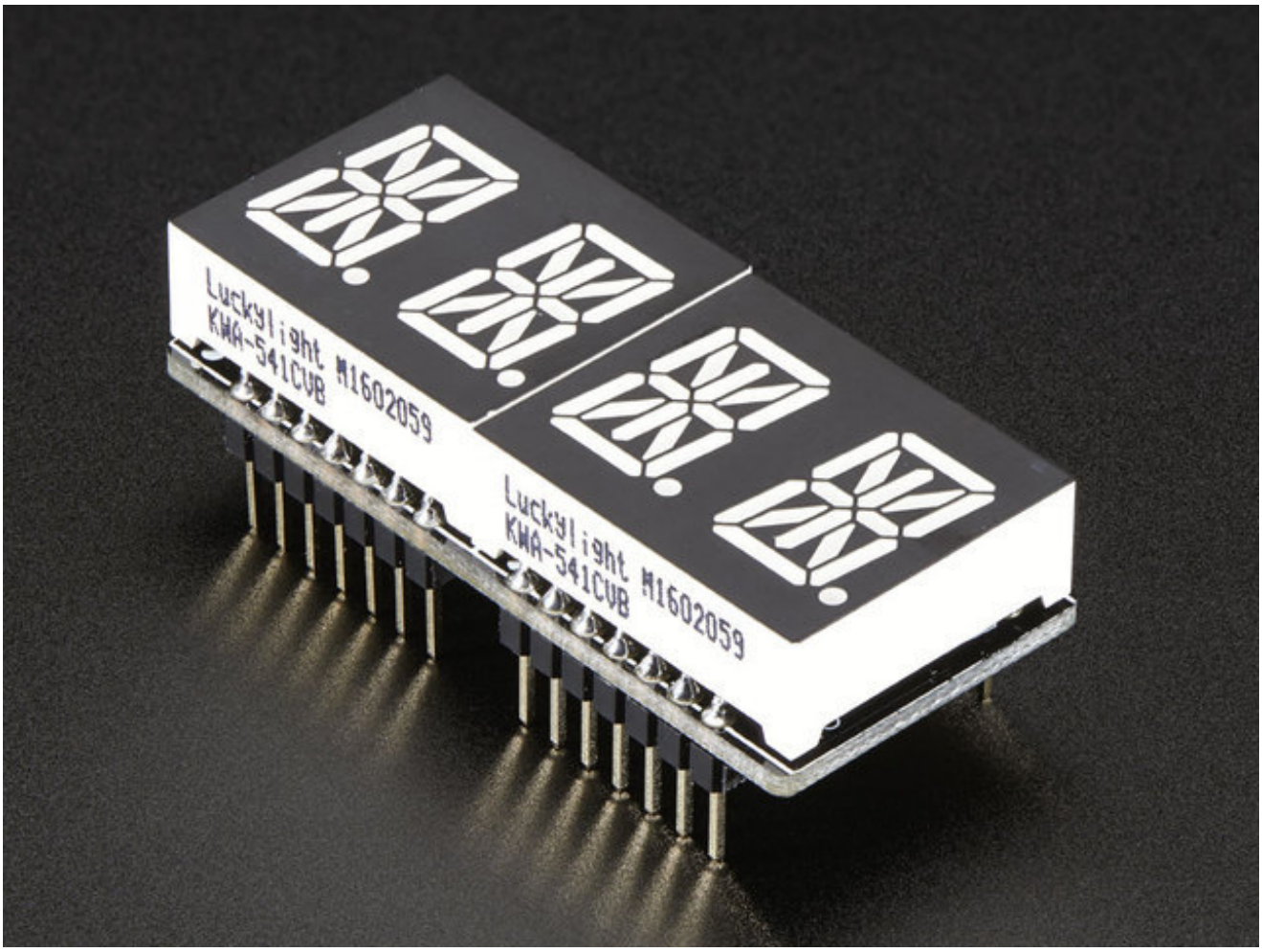
Once soldered, clip each wire short
- 

- 

- 

You're done! You can now install software and get blinking

Usage



To talk to the LED helper chip you'll need to use our Arduino[Adafruit LED Backpack library from github](http://adafru.it/aLI) (http://adafru.it/aLI).

To download you can visit the repository, or simply click on this button:

[Download Adafruit LED Backpack library](http://adafru.it/ncm)
http://adafru.it/ncm

Rename the uncompressed folder **Adafruit_LEDBackpack**. Check that the **Adafruit_LEDBackpack** folder contains **Adafruit_LEDBackpack.cpp** and **Adafruit_LEDBackpack.h** Place the **Adafruit_LEDBackpack** library folder your *arduinosketchfolder*/**libraries**/ folder.

You may need to create the*libraries* subfolder if it's your first library. We also have a great

tutorial on Arduino library installation at:
http://learn.adafruit.com/adafruit-all-about-arduino-libraries-install-use (http://adafru.it/aYM)

Install Adafruit GFX

You will need to do the same for the Adafruit_GFX library available here (http://adafru.it/aJa)

Download Adafruit GFX
http://adafru.it/cBB

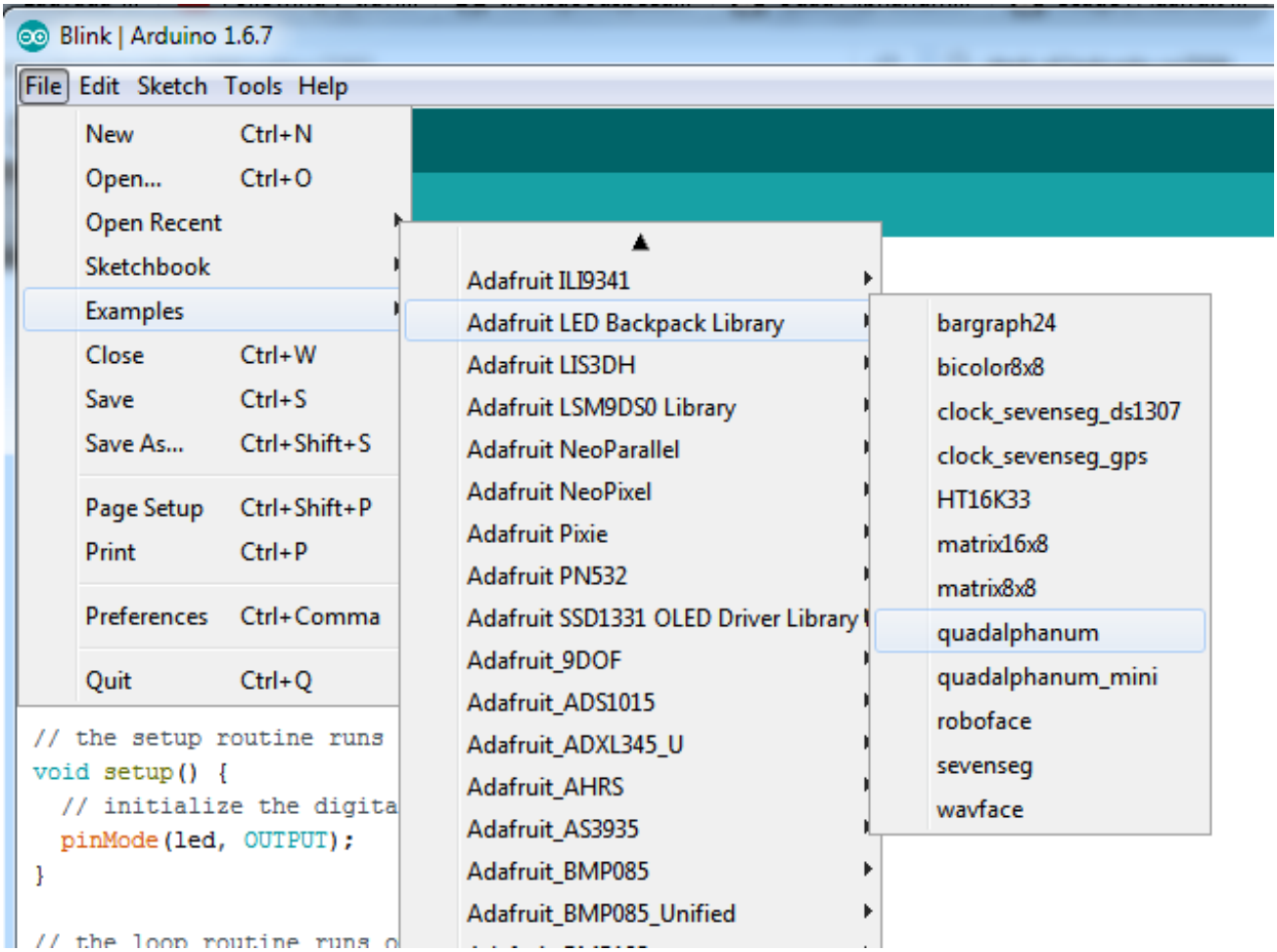
Rename the uncompressed folder **Adafruit_GFX** and check that the **Adafruit_GFX** folder contains **Adafruit_GFX.cpp** and **Adafruit_GFX.h**

Place the **Adafruit_GFX** library folder your ***arduinosketchfolder***/**libraries**/ folder like you did with the LED Backpack library

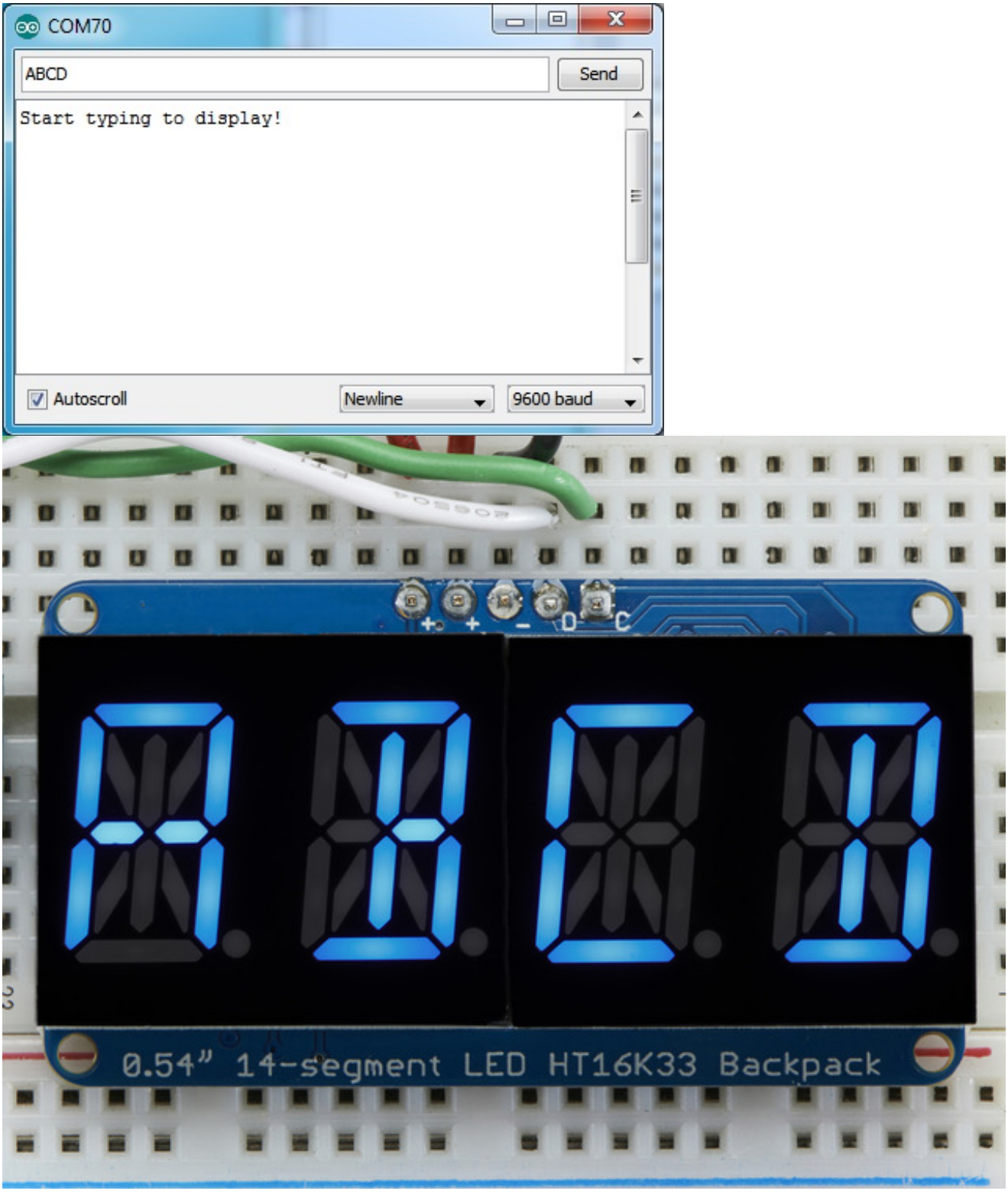
It's not actually used for the 7-segment display, it's only for the matrix backpacks but it's still required by the library so please install it to avoid errors! Restart the IDE.

Run Test!

Once you've restarted you should be able to select the**File->Examples->Adafruit_LEDBackpack->quadalphanum** example sketch. Upload it to your Feather as usual. You should see a basic test program that goes through a bunch of different routines.


Blink | Arduino 1.6.7
File Edit Sketch Tools Help
New Ctrl+N
Open... Ctrl+O
Open Recent
Sketchbook
Examples
Close Ctrl+W
Save Ctrl+S
Save As... Ctrl+Shift+S
Page Setup Ctrl+Shift+P
Print Ctrl+P
Preferences Ctrl+Comma
Quit Ctrl+Q
// the setup routine runs
void setup() {
// initialize the digita
pinMode(led, OUTPUT);
}
Adafruit ILI9341
Adafruit LED Backpack Library
Adafruit LIS3DH
Adafruit LSM9DS0 Library
Adafruit NeoParallel
Adafruit NeoPixel
Adafruit Pixie
Adafruit PN532
Adafruit SSD1331 OLED Driver Library
Adafruit_9DOF
Adafruit_ADS1015
Adafruit_ADXL345_U
Adafruit_AHRS
Adafruit_AS3935
Adafruit_BMP085
Adafruit_BMP085_Unified
bargraph24
bicolor8x8
clock_sevenseg_ds1307
clock_sevenseg_gps
HT16K33
matrix16x8
matrix8x8
quadalphanum
quadalphanum_mini
roboface
sevenseg
wavface

Upload to your Arduino, and open up the Serial console at 9600 baud speed. You'll see each digit light up all the segments, then the display will scroll through the 'font table' showing every character that it knows how to display. Finally, you'll get a notice to start typing into the serial console. Type a message and hit return, you'll see it scroll onto the display!



Library Reference

For the quad displays, we have a special object that can handle ascii data for easy printing.

You can create the object with

```
Adafruit_AlphaNum4 alpha4 = Adafruit_AlphaNum4();
```

There's no arguments or pins because the backpacks use the fixed I2C pins.
By default, the address is 0x70, but you can pass in the I2C address used when you initialize the display with **begin**

```
alpha4.begin(0x70);  // pass in the address
```

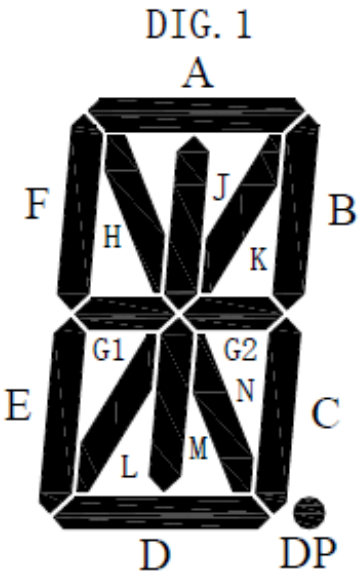
Next up, the segments can be turned on/off for each digit by writing the 'raw' bitmap you want, for example, all the LEDs off on digit #3 is

```
alpha4.writeDigitRaw(3, 0x0);
```

All the segments on for digit #0 is

```
alpha4.writeDigitRaw(0, 0x3FFF);
```

This is the segment map:



the 16 bit digit you pass in for raw image has this mapping:

0 **DP N M L K J H G2 G1 F E D C B A**

The first bit isn't used, you can make it 0 or 1

To turn on just the **A** segment, use 0x0001
To turn on just the **G1** segment, use 0x0040

ASCII data

If you're just looking to print 'text' you can use our font table, just pass in an ASCII character!

For example, to set digit #0 to**A** call:

```
alpha4.writeDigitAscii(0, 'A')
```

Writing Data

Don't forget to 'write' the data to the display with

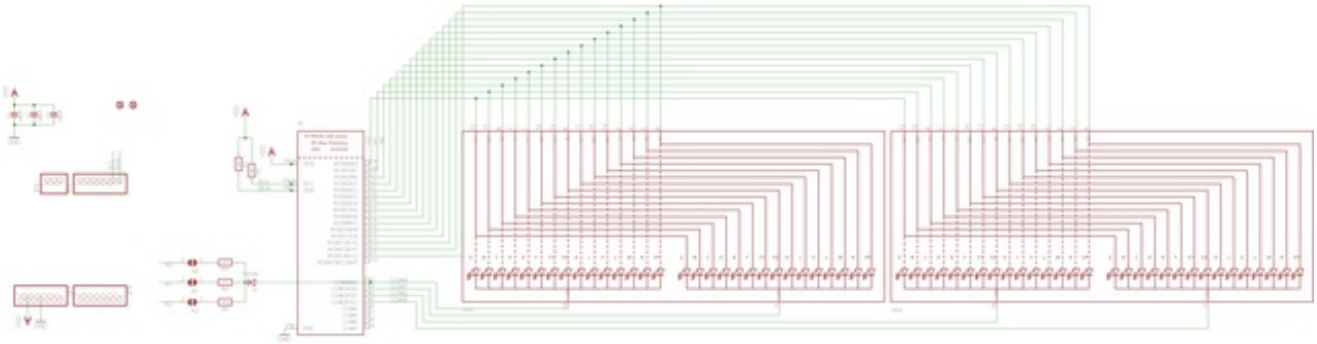
```
alpha4.writeDisplay();
```

That's what actually 'sets' the data onto the LEDs!

Downloads

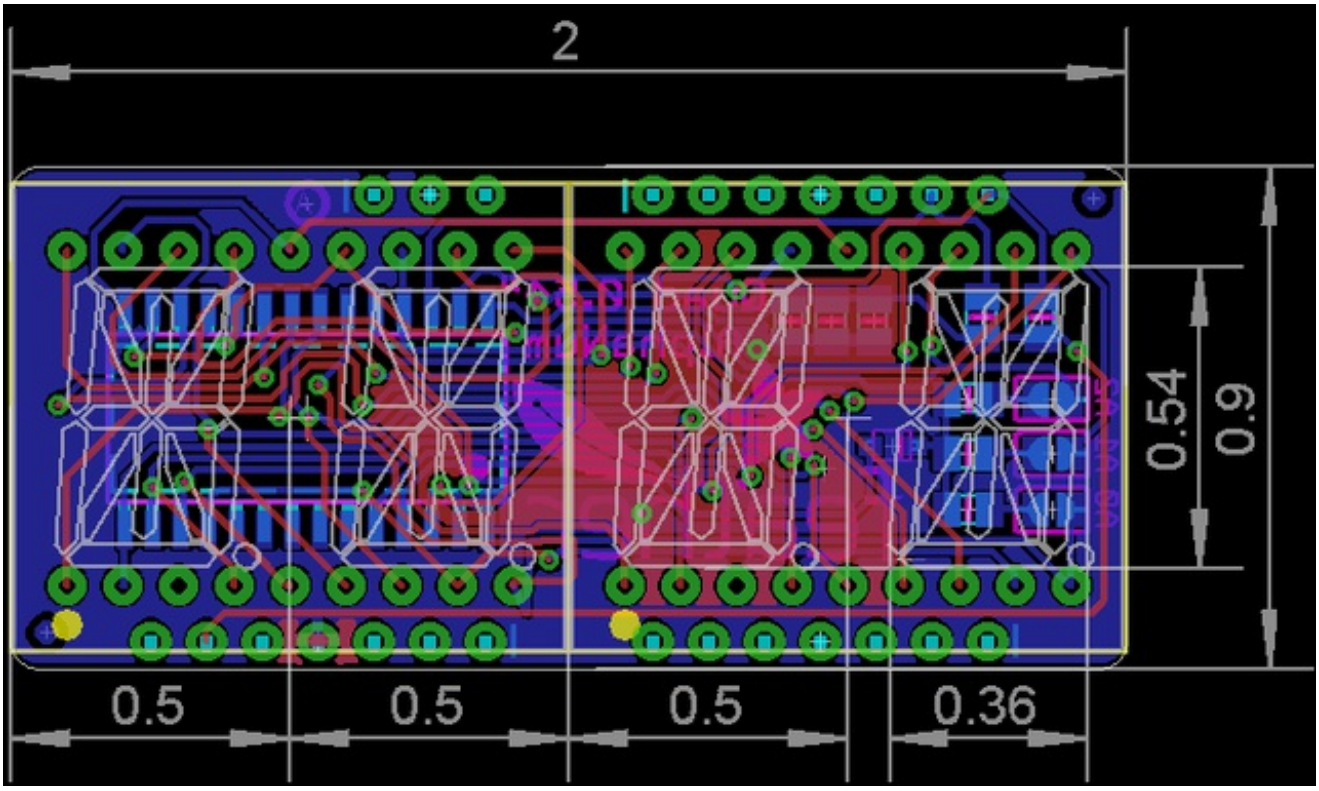
- Arduino LED Backpack Library (http://adafru.it/mau)
- EagleCAD Featherwing PCB files (http://adafru.it/nco)
- Fritzing objects in Adafruit Fritzing library (http://adafru.it/aP3)

Schematic



Fabrication Print

Dimensions in inches


2
0.54
0.9
0.5
0.5
0.5
0.36