



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

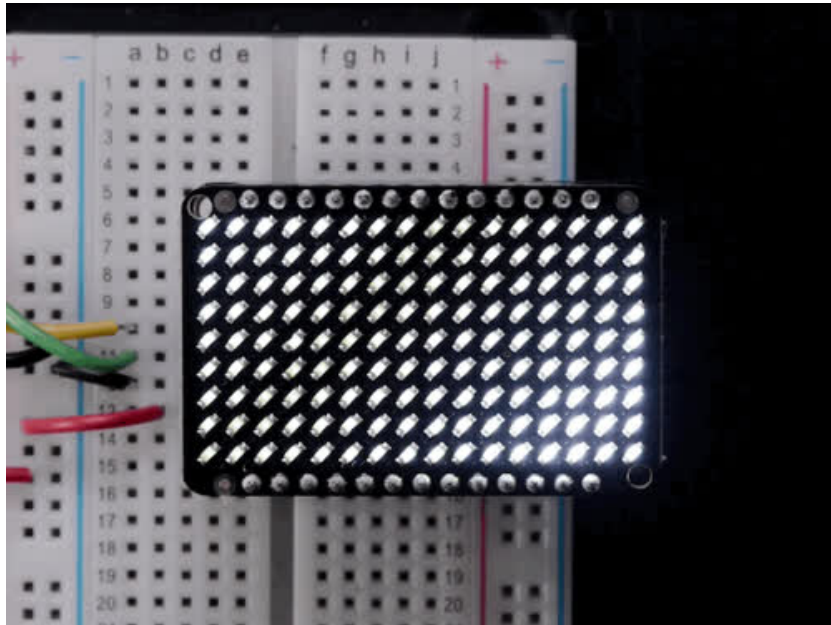
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



□

IS31FL3731 16x9 Charlieplexed PWM LED Driver

Created by lady ada

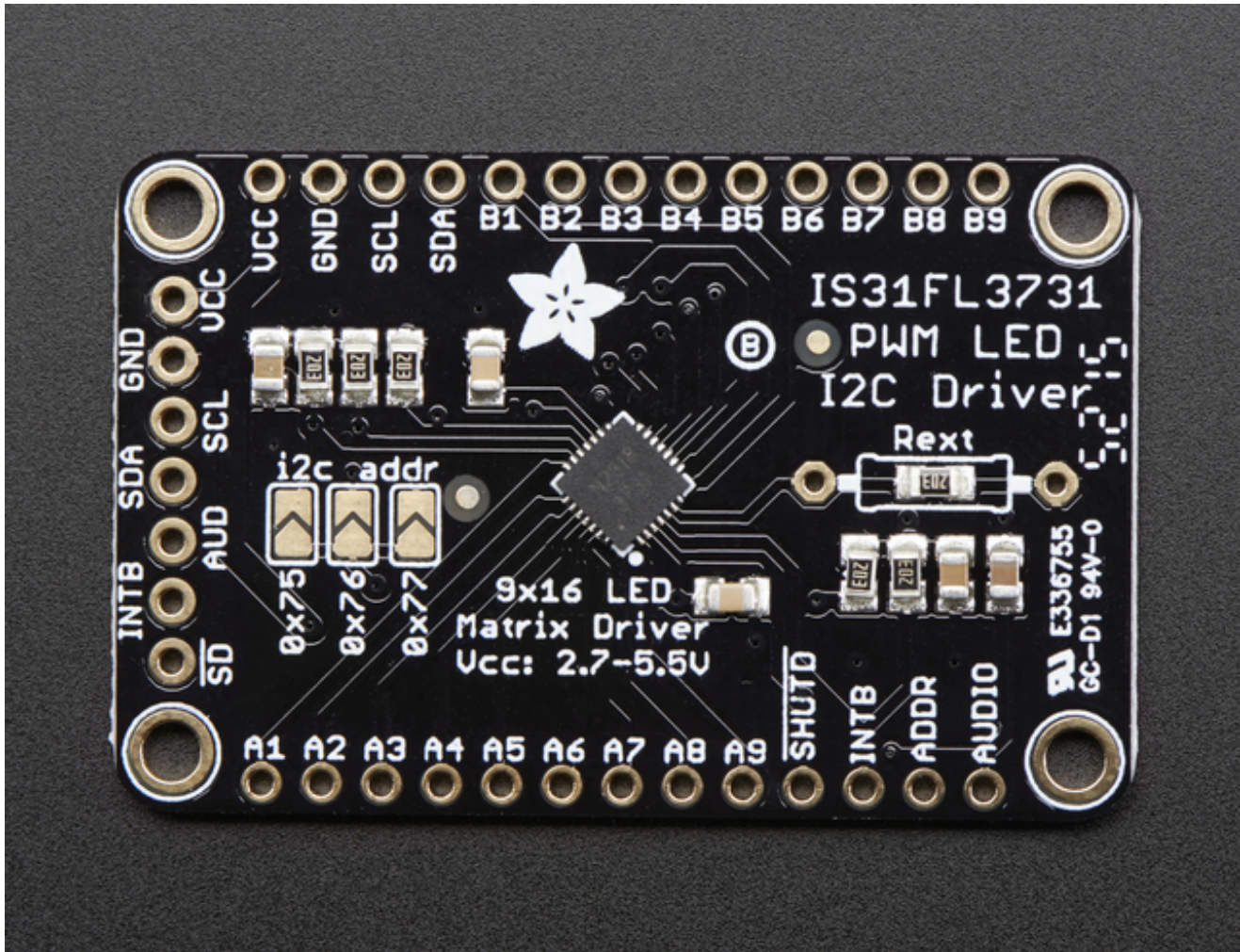


Last updated on 2016-09-10 12:56:50 AM UTC

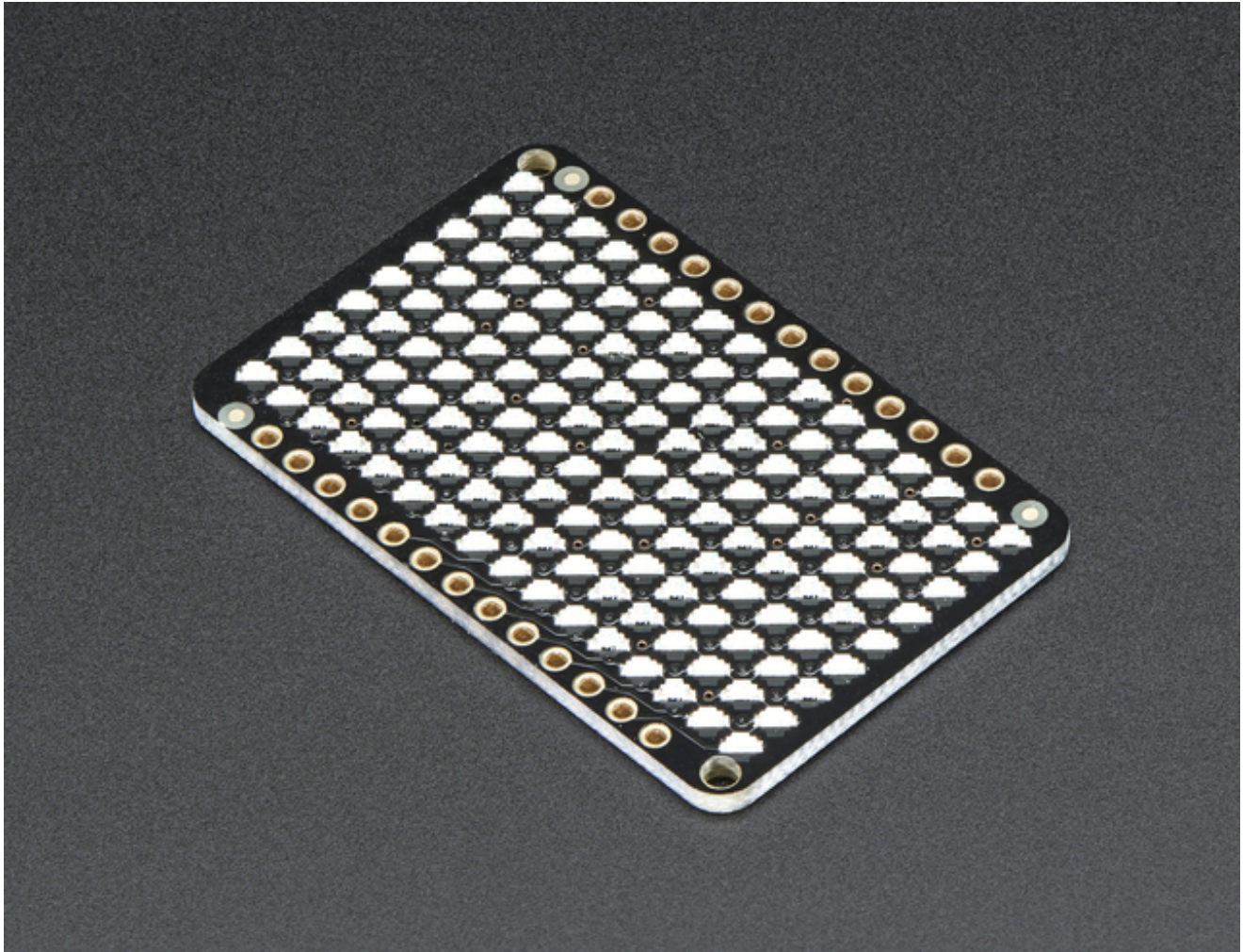
Guide Contents

Guide Contents	2
Overview	3
Pinouts	7
Power Pins	7
I2C Data Pins	8
Other Control Pins	9
LED Grid	10
Assembly	12
Assemble the IS31 Driver Board	12
Prepare the header strip:	13
Add the breakout board:	13
And Solder!	14
Solder Driver Headers for LEDs	15
Prepare the header strip:	15
Add the breakout board face up:	16
And Solder!	17
Attach LED panel	18
Wiring & Test	24
Wiring	24
Download Adafruit_IS31FL3731 library	25
Install Adafruit GFX	25
Load Demo	25
Library Reference	27
Initialize	27
Drawing	27
Adafruit GFX	27
Multiple Buffers	28
Downloads	29
Datasheets & Files	29
IS31FL3731 Breakout	29
16x9 0603 LED Grid	30

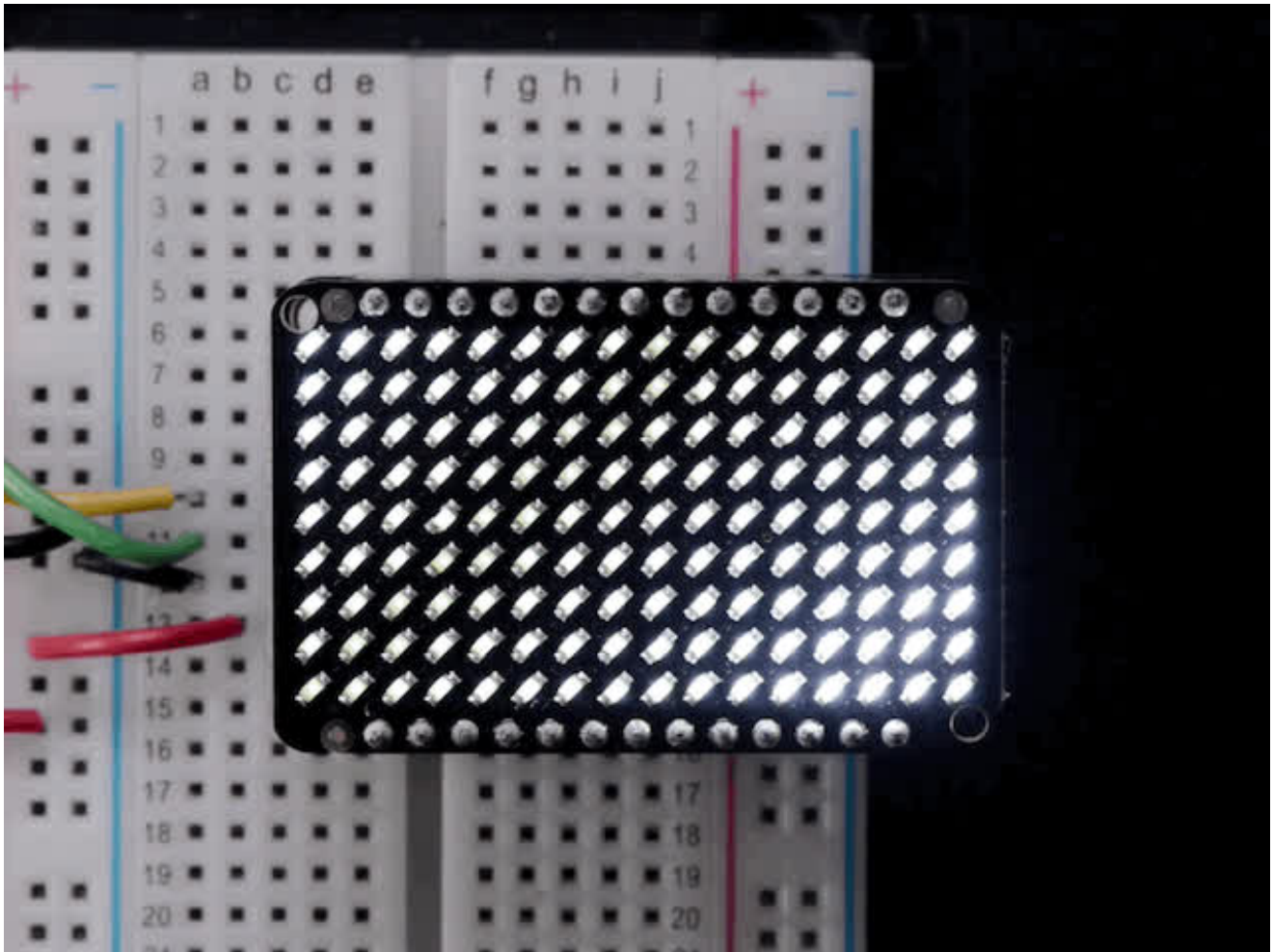
Overview



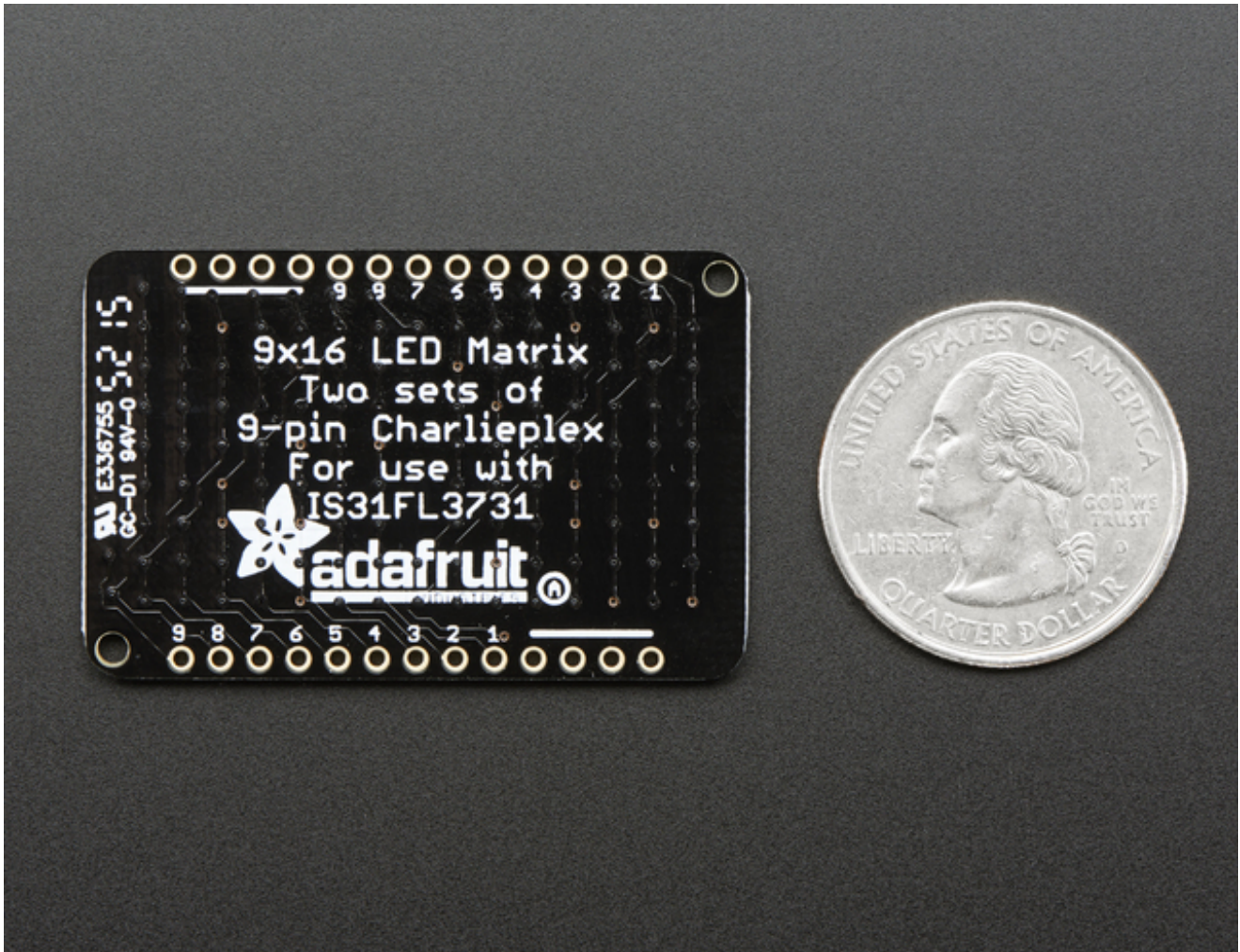
The IS31FL3731 will let you get back to that classic LED matrix look, with a nice upgrade! **This I2C LED driver chip has the ability to PWM each individual LED in a 16x9 grid** so you can have beautiful LED lighting effects, without a lot of pin twiddling. Simply tell the chip which LED on the grid you want lit, and what brightness and it's all taken care of for you.



The IS31FL3731 is a nice little chip - it can use 2.7-5.5V power and logic so its flexible for use with any microcontroller. You can set the address so up to 4 matrices can share an I2C bus. Inside is enough RAM for 8 seperate frames of display memory so you can set up multiple frames of an animation and flip them to be displayed with a single command.

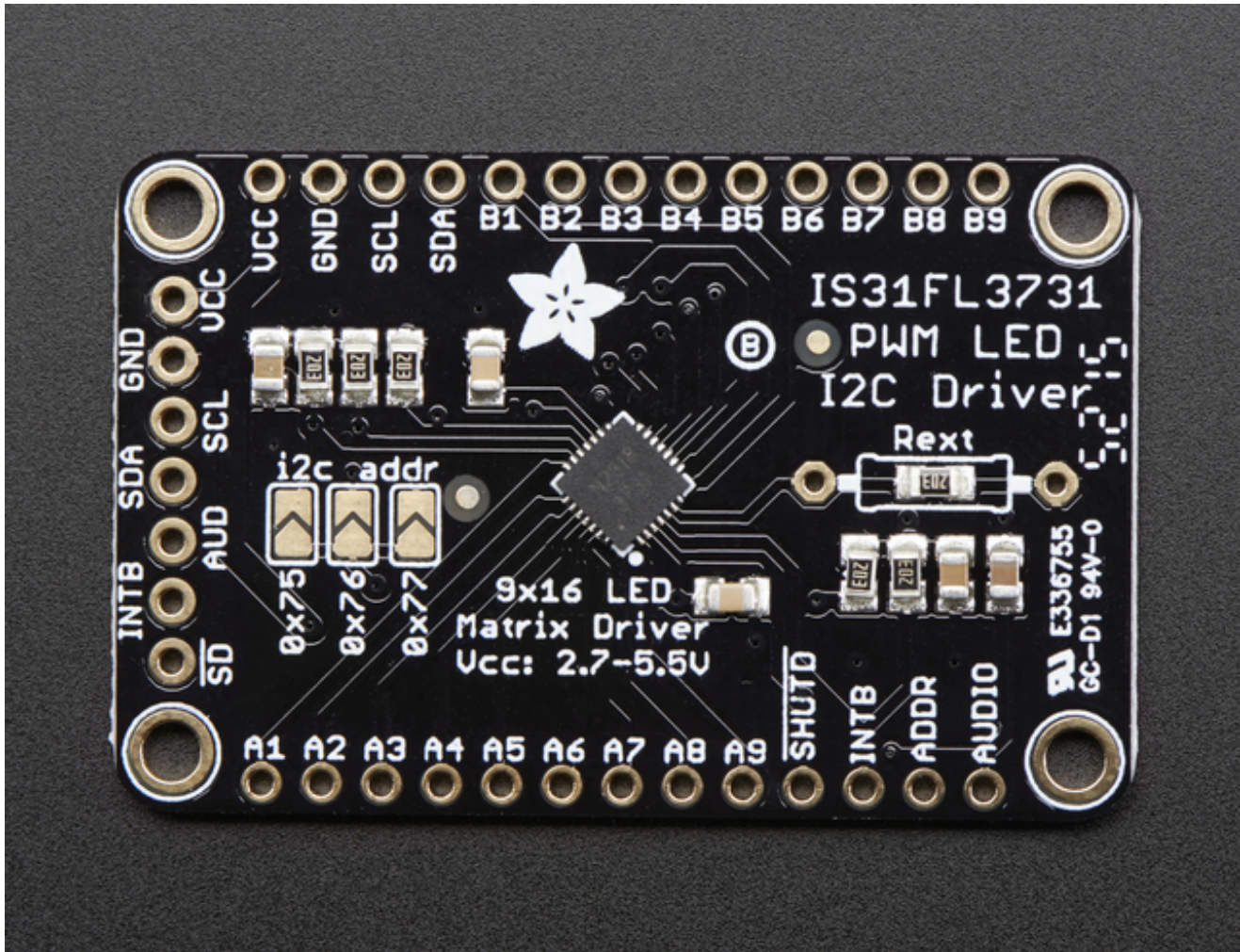


This chip is great for making small LED displays, and we even designed the breakout to match up with our ready-to-go LED grids in red, yellow, green, blue and white. Sandwich the driver and matrix breakout, solder together for a compact setup. *Or* you can DIY your own setup, just follow the LED grid schematic in the IS31FL3731 datasheet.



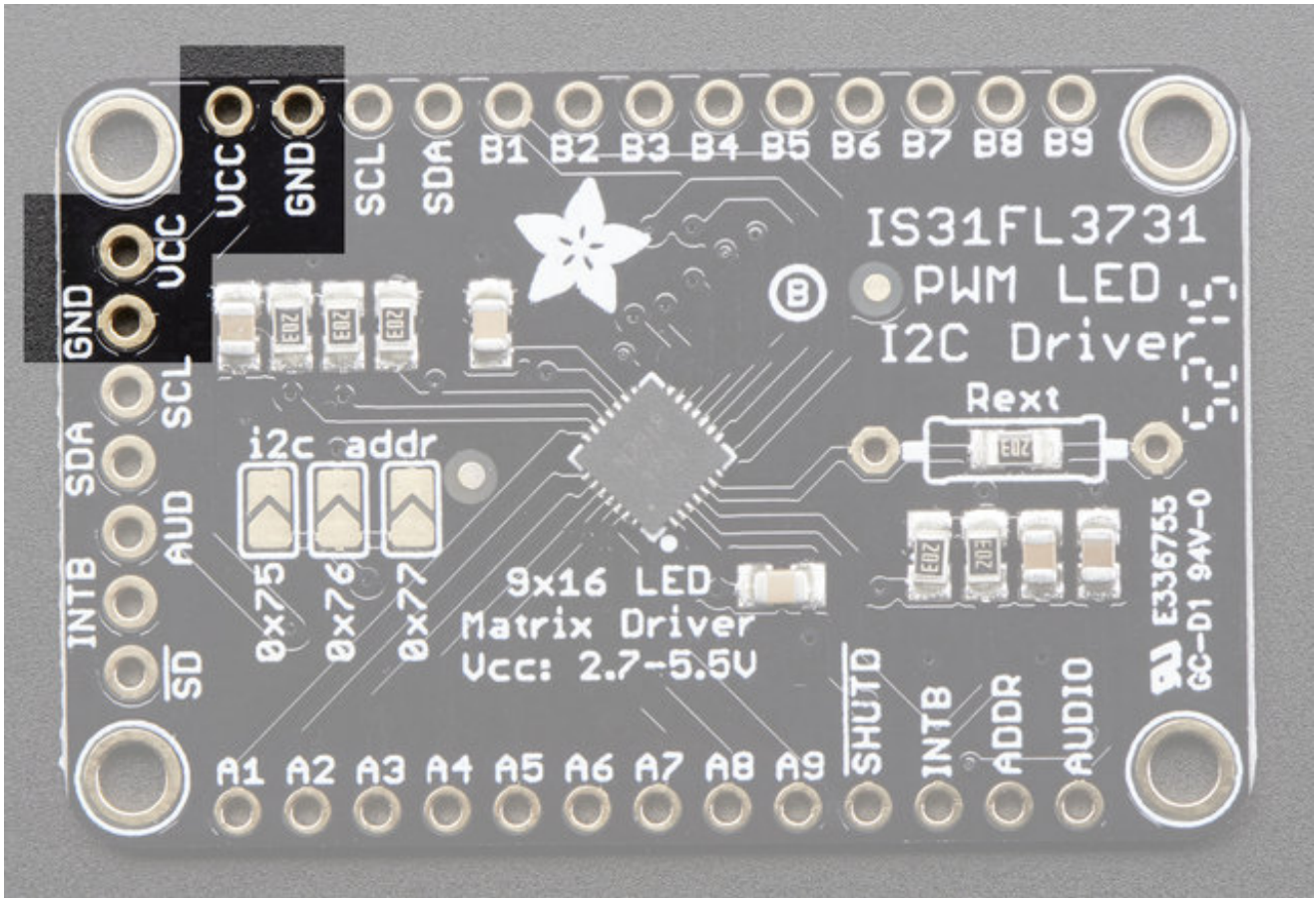
Pick up a driver board and your favorite color LEDs to match. You'll need to do some basic soldering to attach the driver backpack and matrix together, and run wires to your microcontroller, but its not too hard. Then install our Arduino code to get some LEDs blinking immediately. Our library is Adafruit_GFX compatible so you can draw lines, circles, text, and small bitmaps if you want more graphics control

Pinouts



The IS31FL3731 has a lot of pins, and we wanted to make it easy to use with a breadboard while sandwiched with an LED matrix. The easiest way we could figure out to do this is make the board as large as our 0603-LED 16x9 matrix grids and have a control header on one edge. That way you can solder the two long headers directly to the matrix and still have access to pins for power and data.

Power Pins

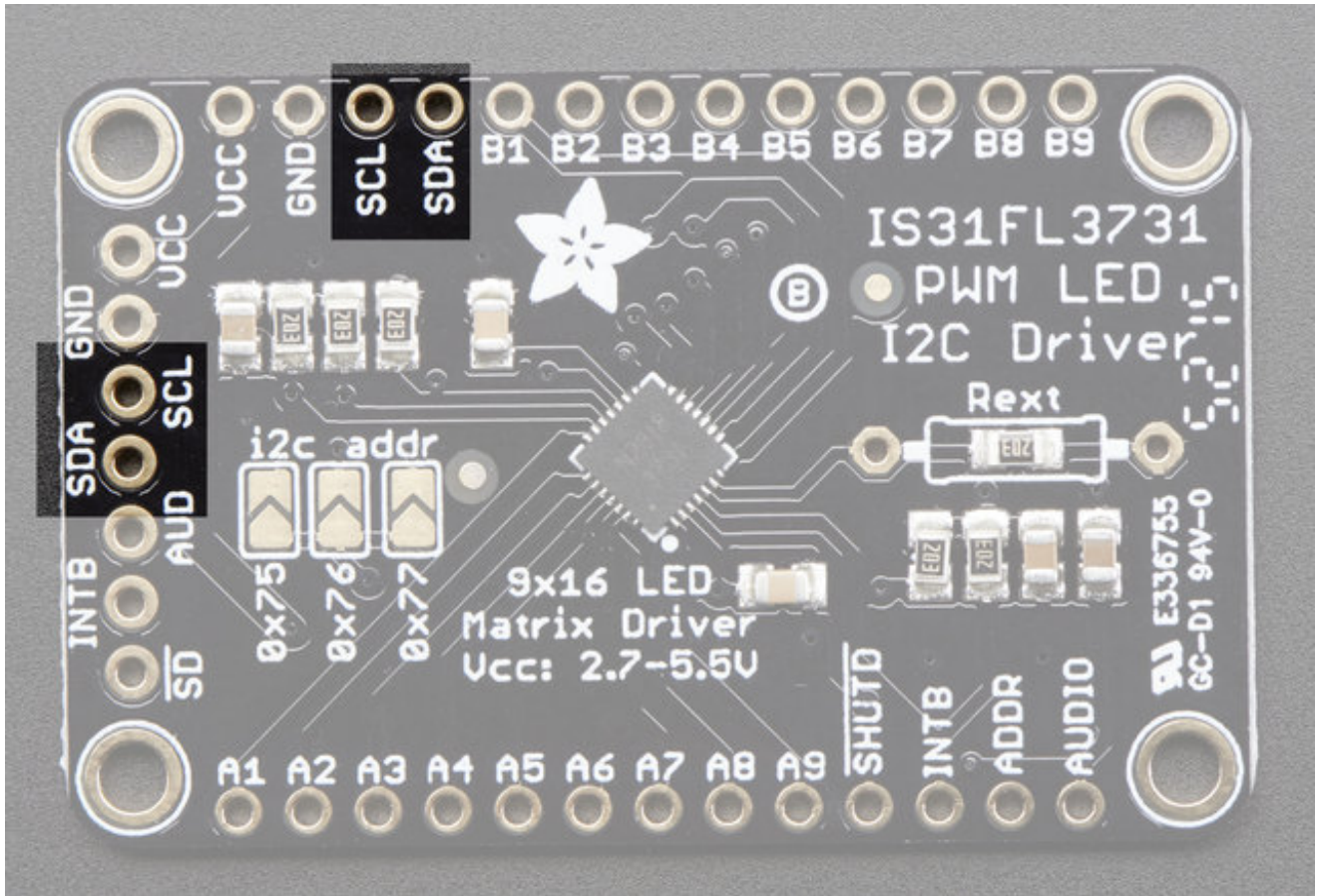


You can power the IS31 from 2.7-5.5VDC, but note that the same voltage is used for both power and logic.

If you are using a **5V** logic device, just connect **VCC** to 5V.

If you are using a **3.3V** logic, you can either power with 3.3V, which will work fine for red, yellow or light green LEDs *or* you can power from 5V and then use ~2.2K resistors from SDA and SCL to 3.3V to 'overpower' the built in 20K pullup resistors.

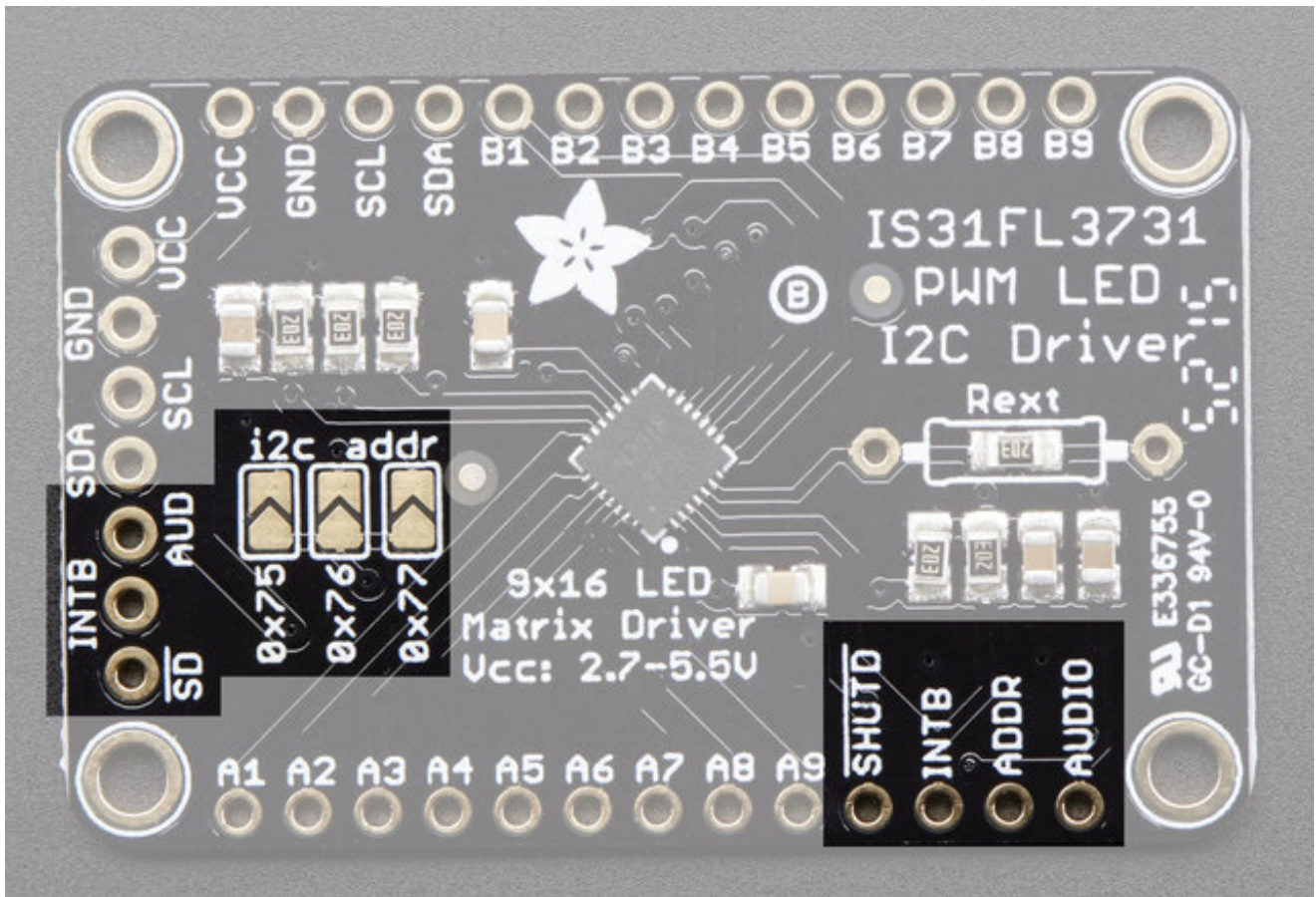
I2C Data Pins



This chip uses I2C for control, it does not use clock stretching or repeated start. There are built in 20K pullups to VCC. You can run it as fast as 400KHz clock speed, but you may need to add additional 2K pullups from SDA and SCL up to VCC for higher speeds

- **SDA** - I2C data line, connect to your microcontroller's I2C SDA pin
- **SCL** - I2C clock line, connect to your microcontroller's I2C SCL pin

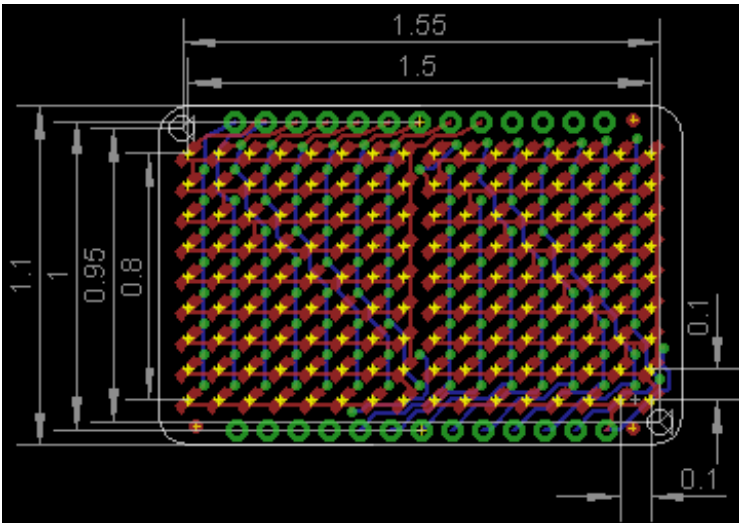
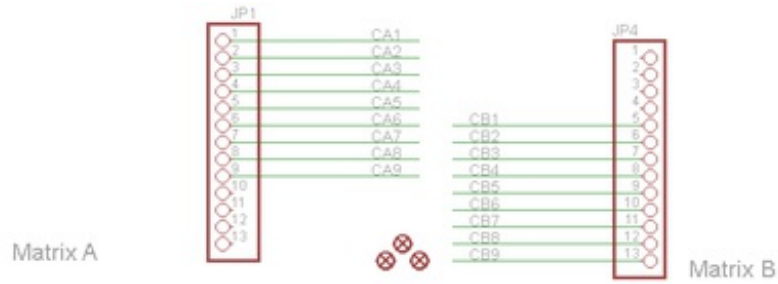
Other Control Pins

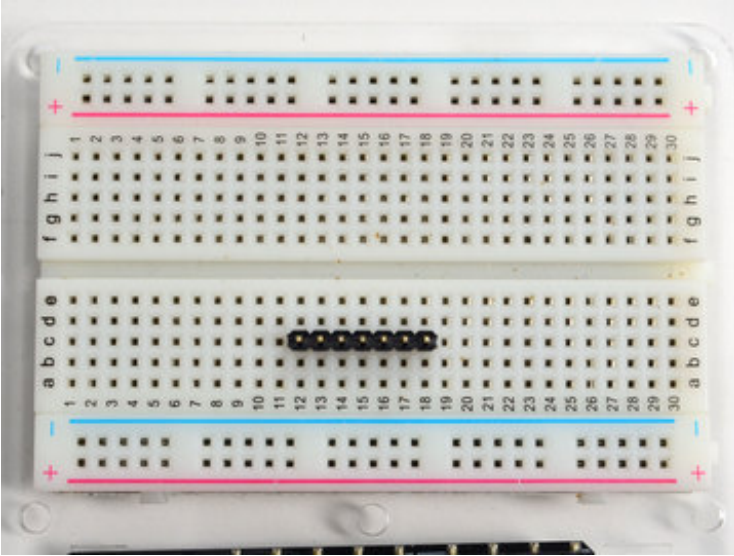


- **SD** - Shutdown pin, default pulled up to VCC. Connect to ground to put the chip in shutdown mode
- **AUD** - Audio input, can be used to modulate the entire display with the amplitude of a line level audio signal, has a series capacitor installed.
- **ADDR** and **jumper**s - By default the address is **0x74** but you can close **one** of the jumpers to change the address for up to 4 devices with varying addresses
- **INTB** - Output interrupt from chip when using the built in animation modes

LED Grid

The LED Grid is much simpler, it just has 2 charlieplex grids, 16x9 total 0603 LEDs, with the two grids broken out to side pins that line up with the driver

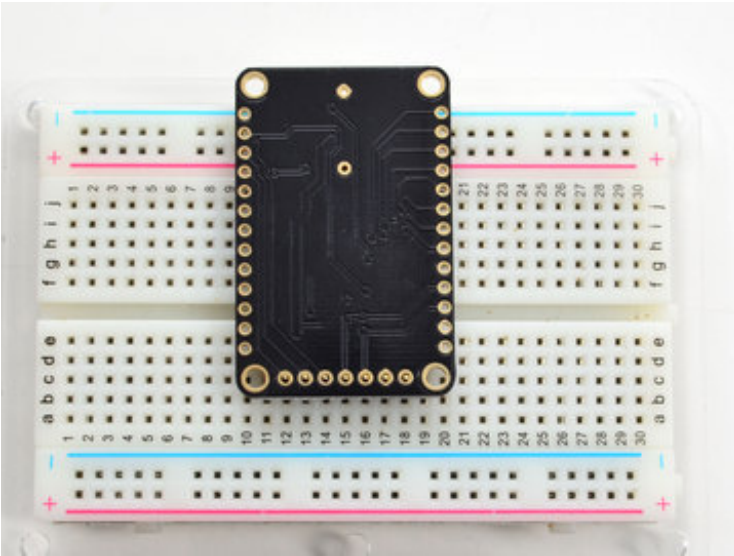




Prepare the header strip:

Cut the strip to length if necessary. It will be easier to solder if you insert it into a breadboard - **long pins down**

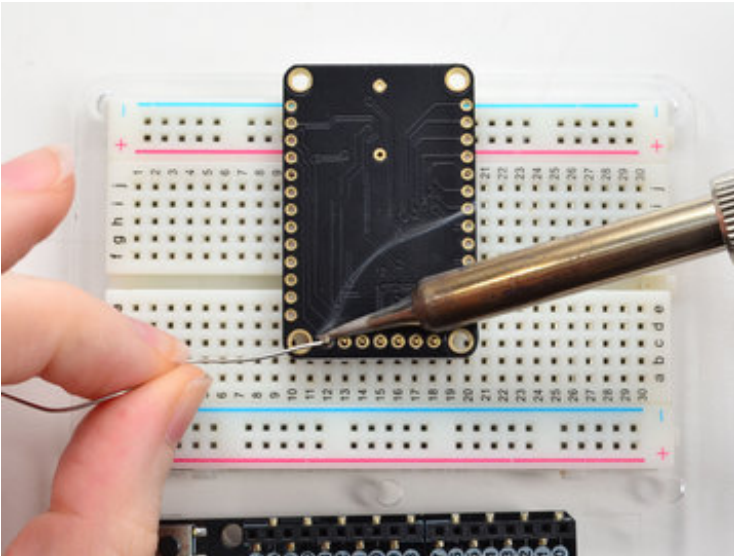
•



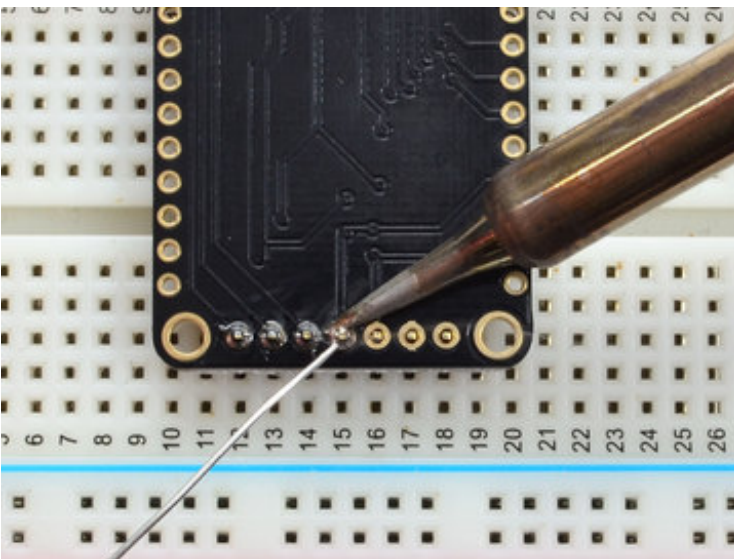
Add the breakout board:

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

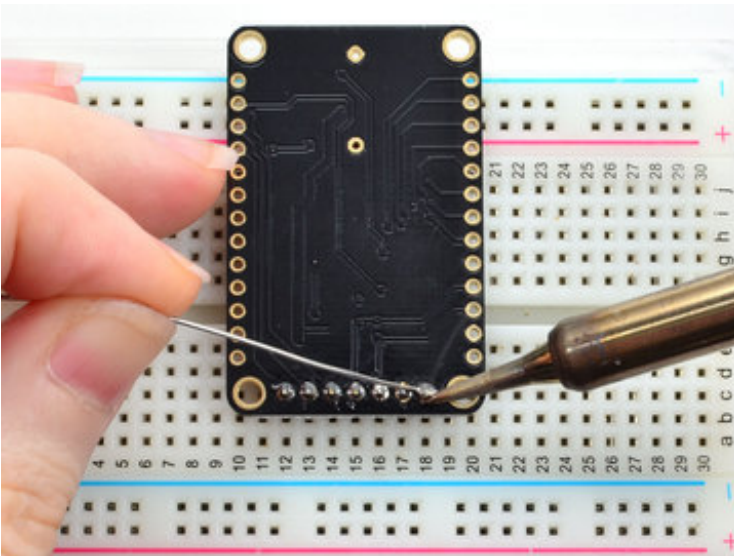
•



•



•

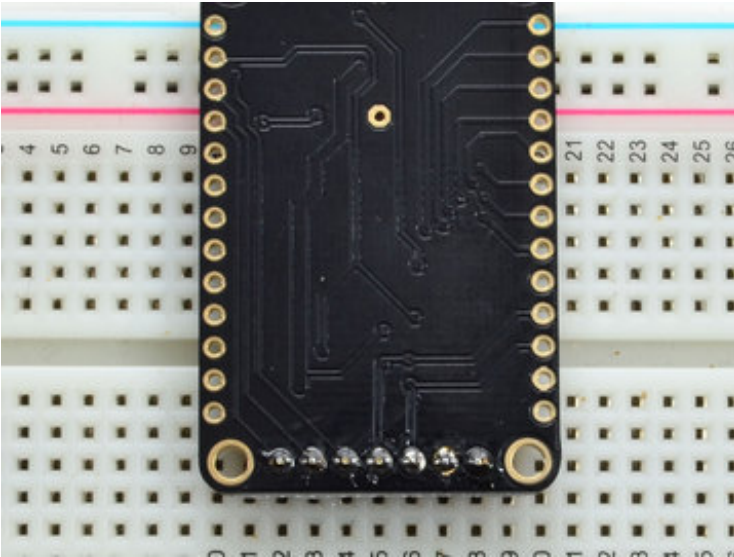


•

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://adafruit.it/aTk) (<http://adafruit.it/aTk>)).

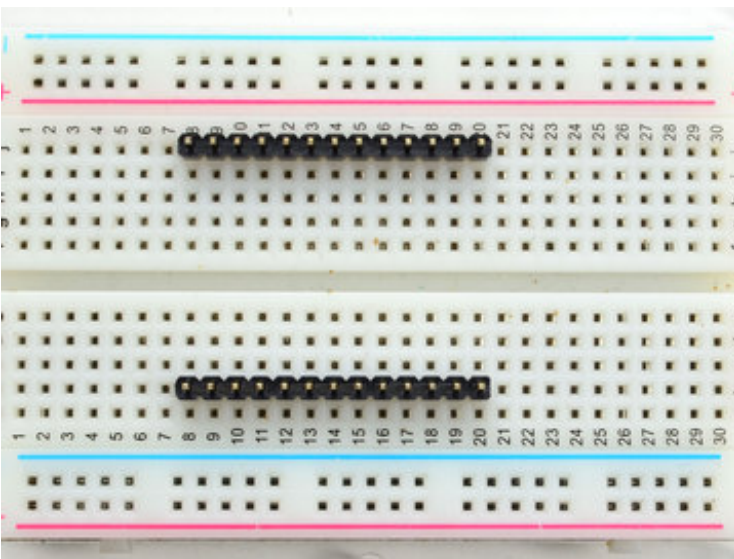


OK the control port header is done.

Check your solder joints visually and continue onto the next steps

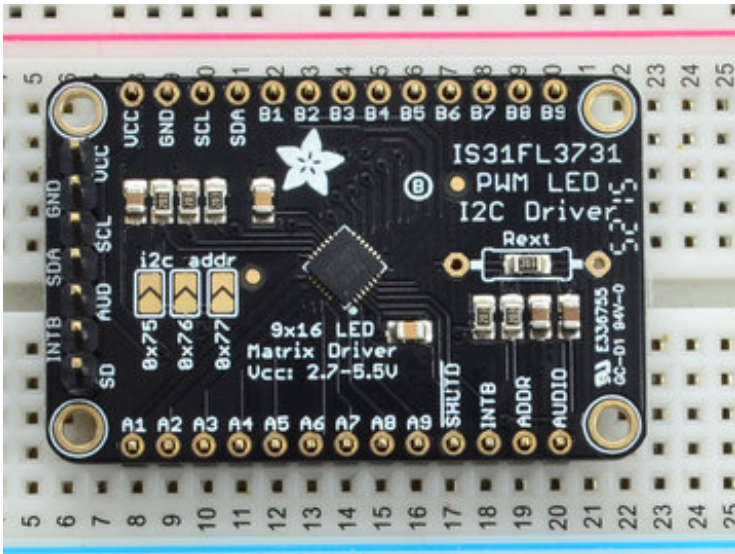
Solder Driver Headers for LEDs

The two side strips are what are used to control the charlie-plexed LEDs



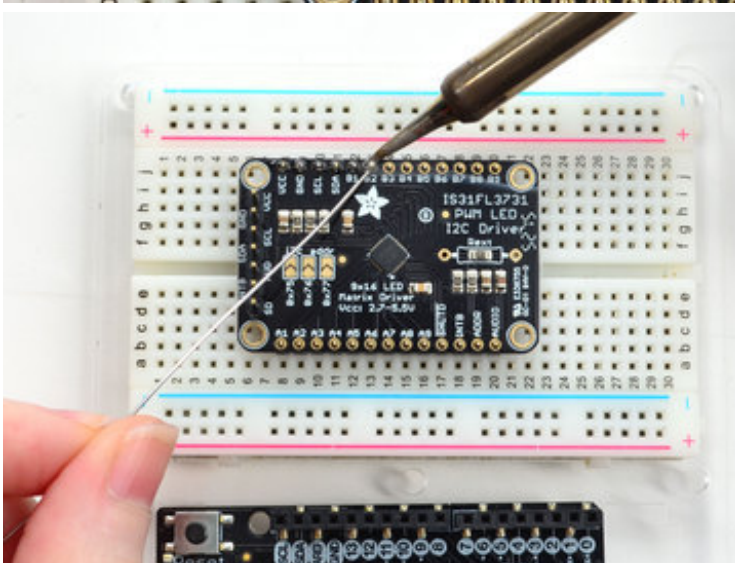
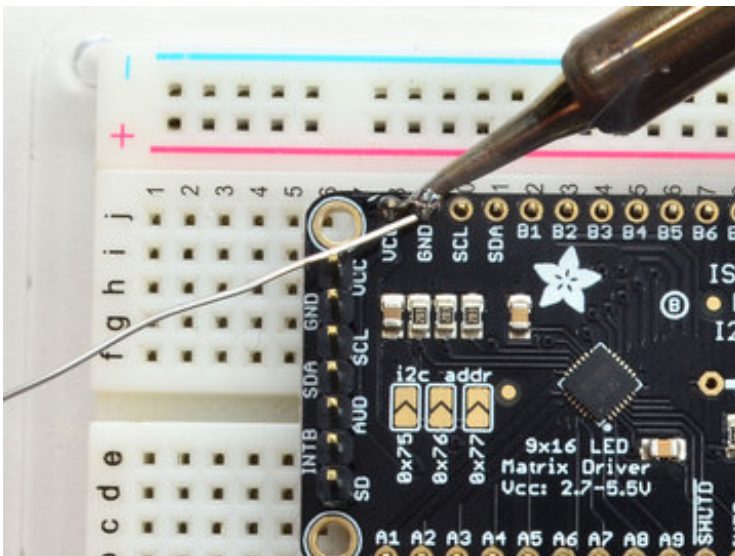
Prepare the header strip:

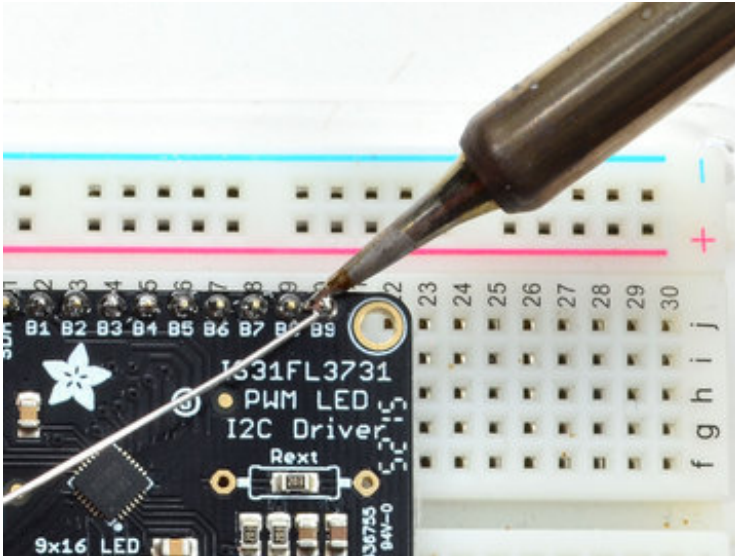
Cut the strip to length if necessary. It will be easier to solder if you insert it into a breadboard - **long pins down**



Add the breakout board face up:

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

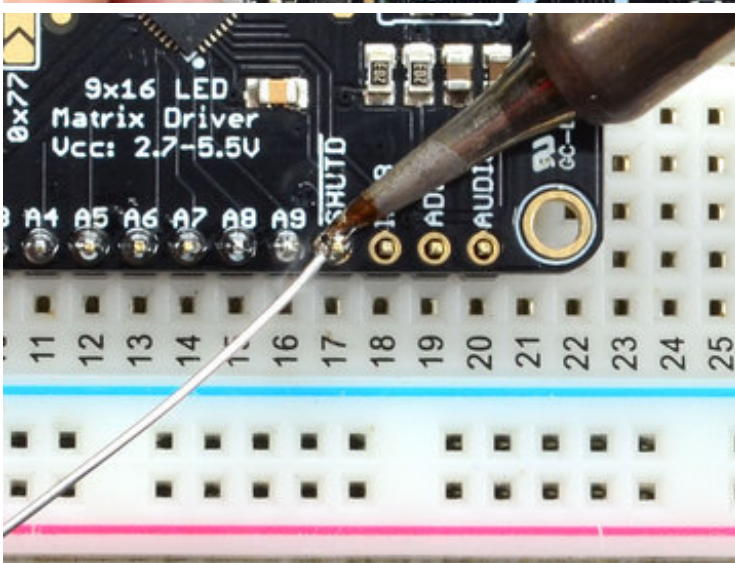
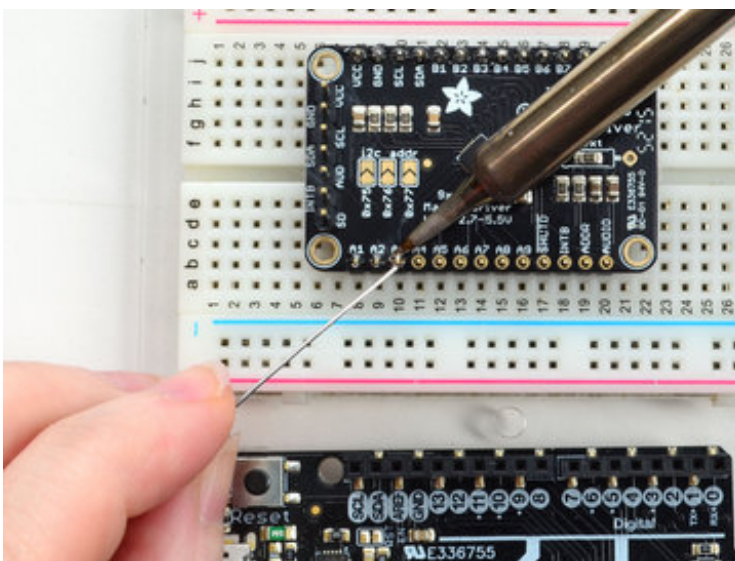


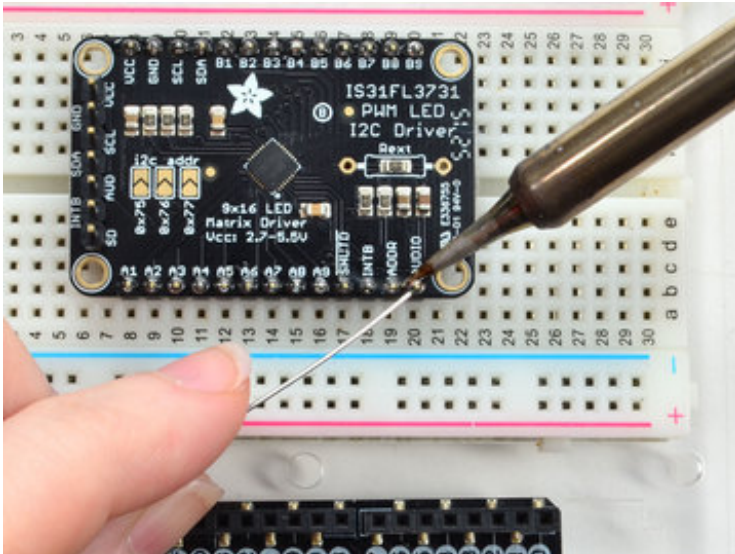


And Solder!

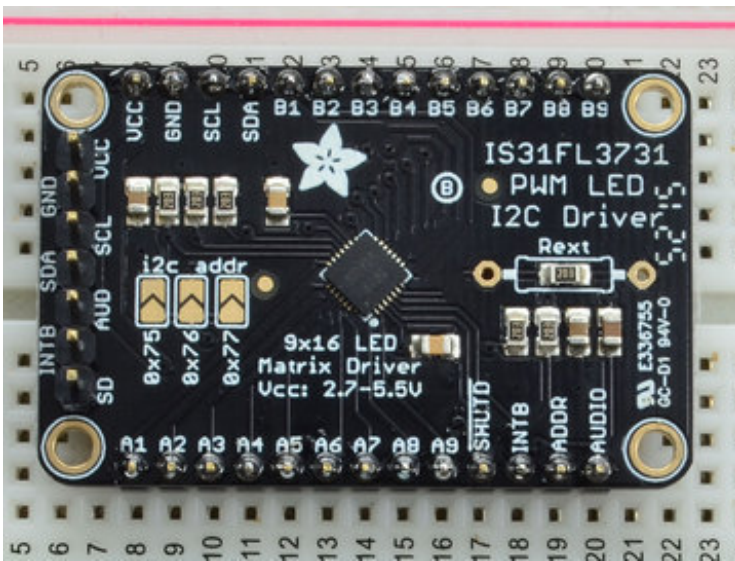
Be sure to solder all pins for reliable electrical contact.

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





•



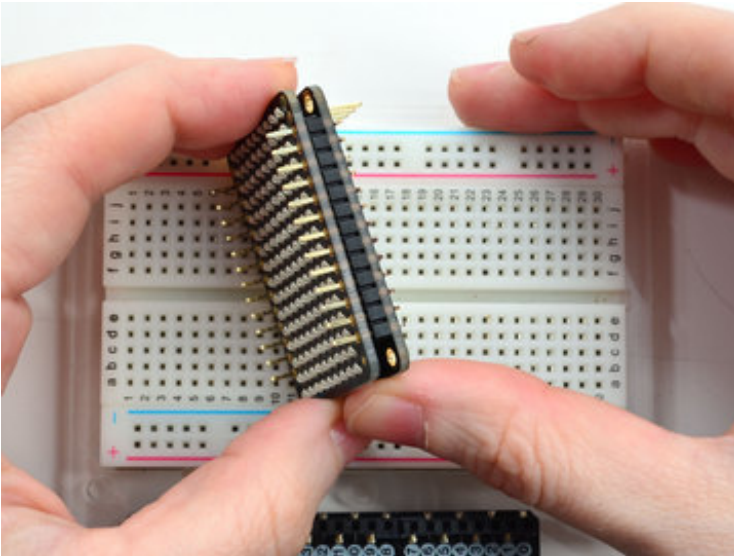
•

OK now you have the control *and* LED pads with headers.

Check your solder joints visually and continue onto the next steps

Attach LED panel

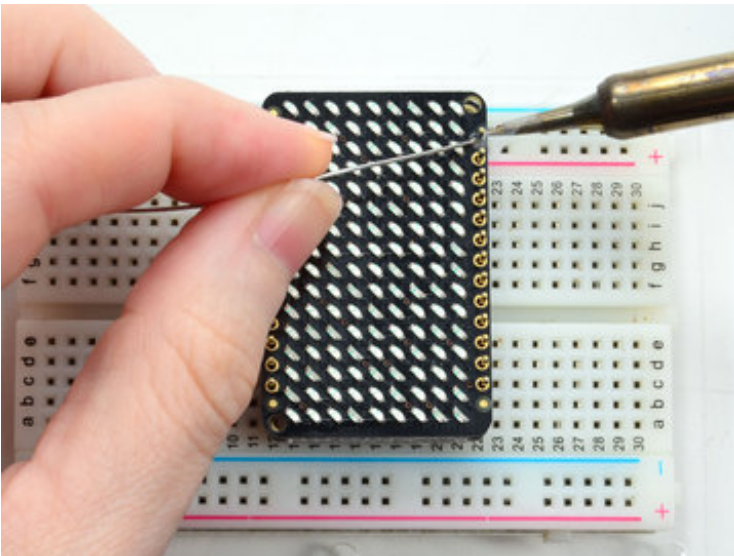
Now we'll sandwich on the charlieplexed LED panel



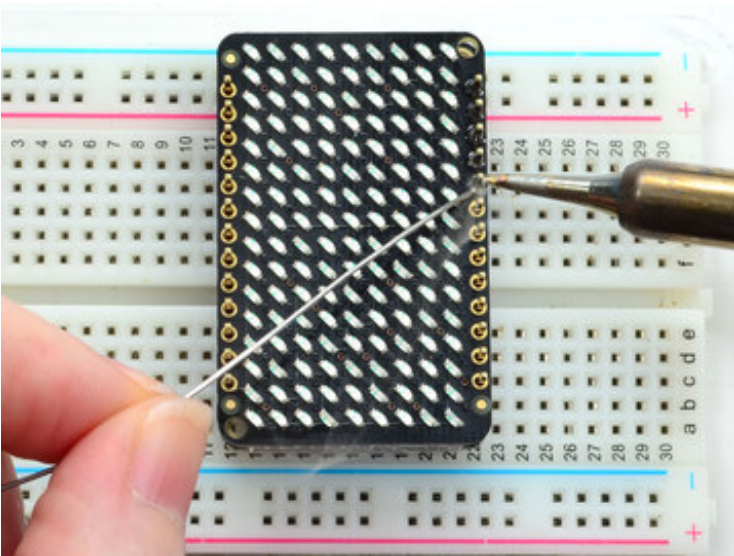
The LEDs face out and connect to the two side header strips.

The panel is **symmetric** - you can flip it around either way and it will work fine

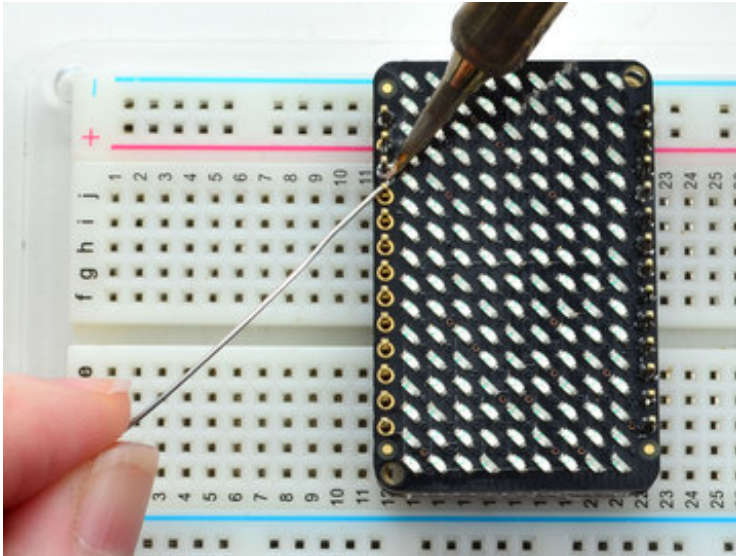
•



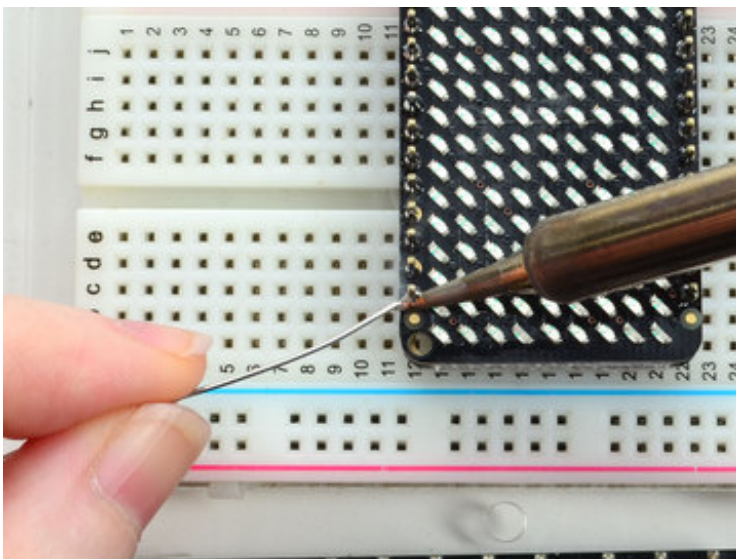
•



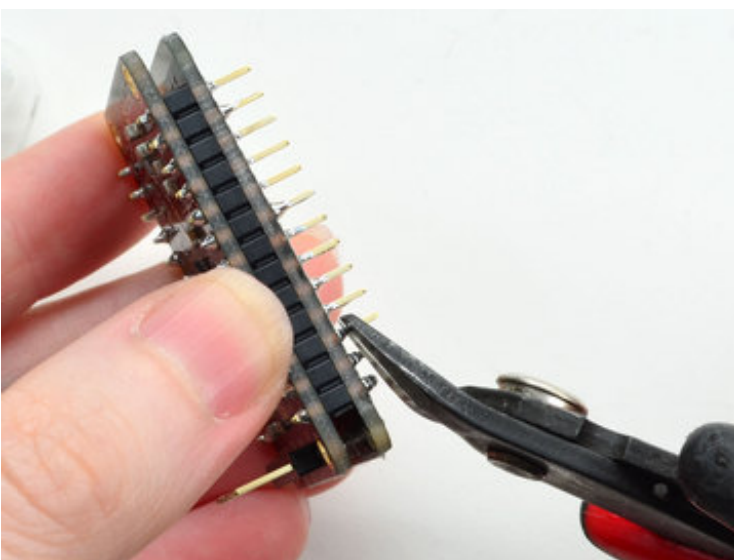
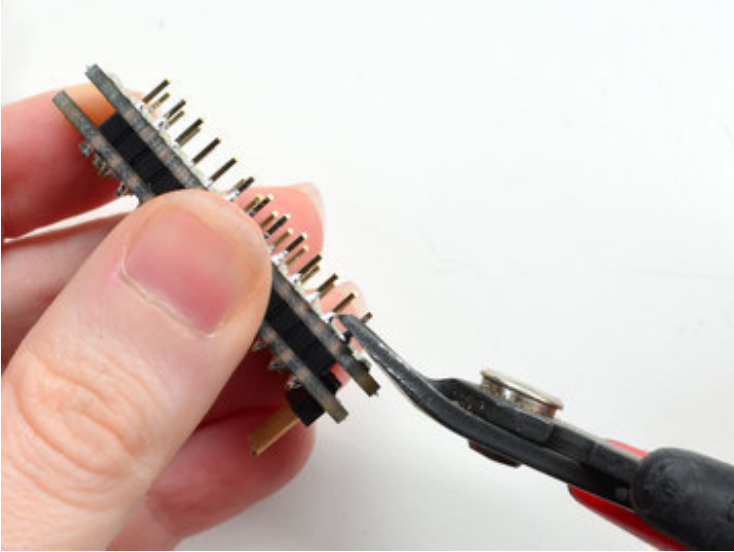
•



Solder in the two long header strips. Not all are connected on the LED matrix but it makes the connections mechanically stable.

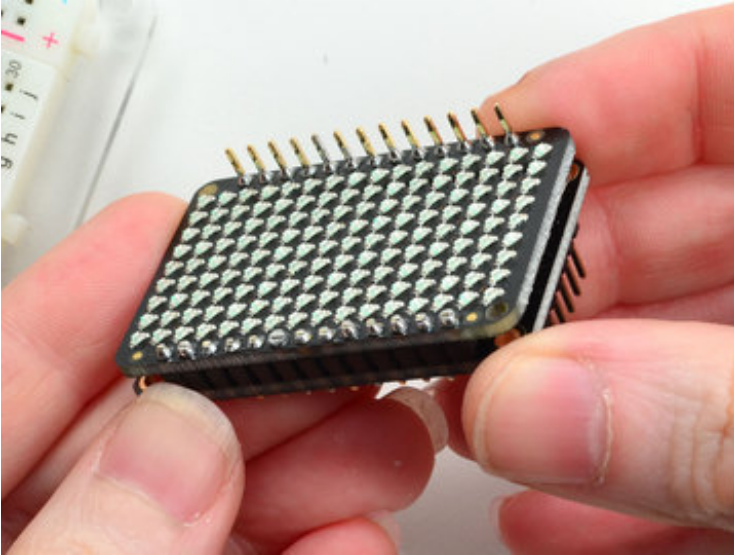


Check your solder joints visually and continue onto the next steps

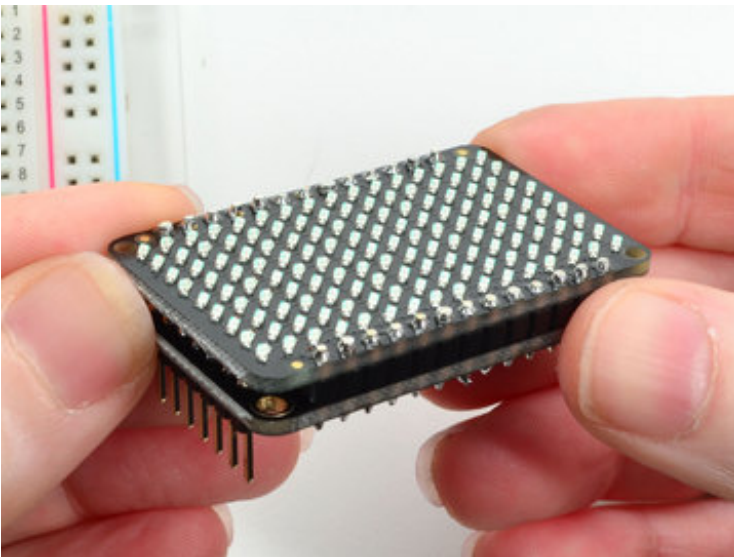


Now you can trim the long header with diagonal cutters. Cut one or two pins at a time

Watch out for flying header bits. Wear eye protection and be careful!

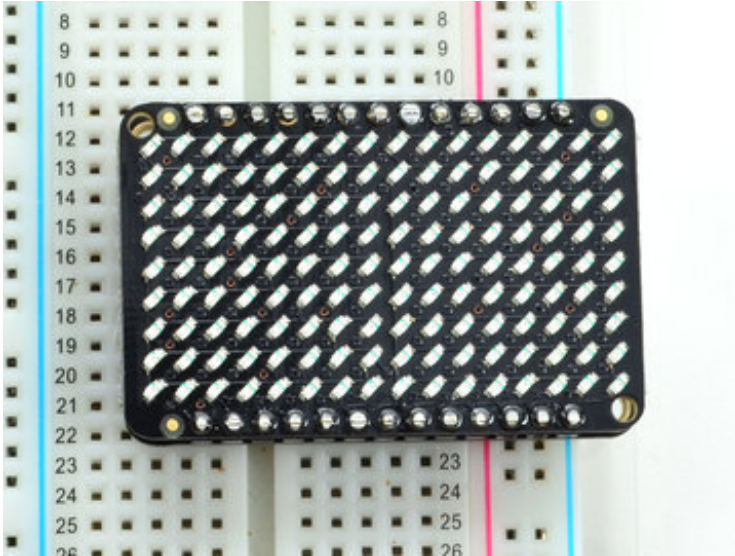


•



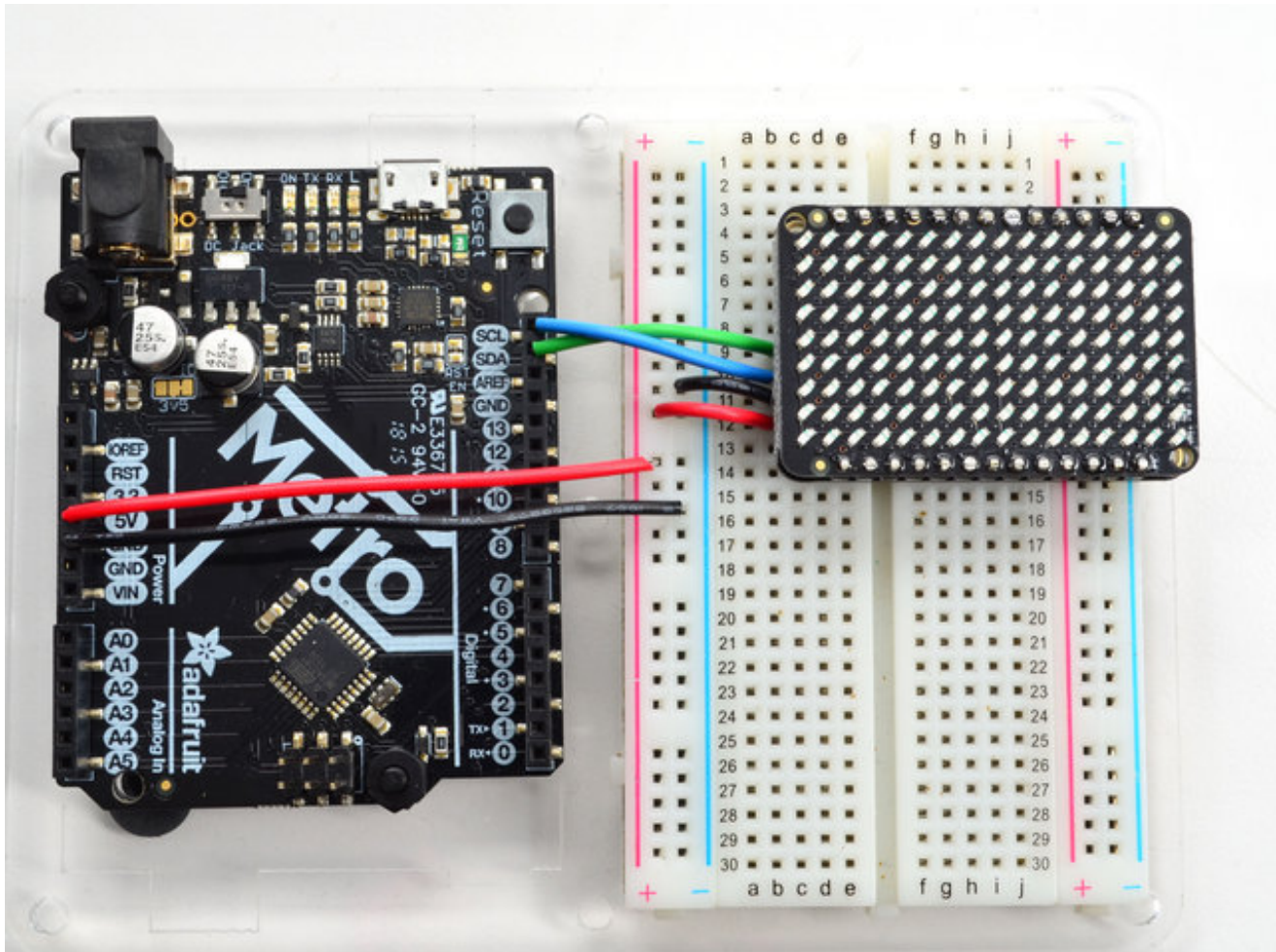
•

Cut both sides down for the best look



You're done! Check your solder joints visually and continue onto the next steps

Wiring & Test



You can easily wire this breakout to any microcontroller, we'll be using an Arduino. For another kind of microcontroller, as long as you have I2C pins available, just check out the library, then port the code.

Wiring

Use this wiring if you want to connect via I2C interface

- Connect **Vin** to the power supply, 3-5V is fine. Use the same voltage that the microcontroller logic is based off of. For most Arduinos, that is 5V
- Connect **GND** to common power/data ground
- Connect the **SCL** pin to the I2C clock **SCL** pin on your Arduino. On an UNO & '328

based Arduino, this is also known as **A5**, on a Mega it is also known as **digital 21** and on a Leonardo/Micro, **digital 3**

- Connect the **SDA** pin to the I2C data **SDA** pin on your Arduino. On an UNO & '328 based Arduino, this is also known as **A4**, on a Mega it is also known as **digital 20** and on a Leonardo/Micro, **digital 2**

Download Adafruit_IS31FL3731 library

To begin reading sensor data, you will need to [download Adafruit_IS31FL3731 from our github repository \(http://adafru.it/IWA\)](http://adafru.it/IWA). You can do that by visiting the github repo and manually downloading or, easier, just click this button to download the zip

[Download the Adafruit_IS31FL3731 Library](http://adafru.it/IWB)

<http://adafru.it/IWB>

Rename the uncompressed folder **Adafruit_IS31FL3731** and check that the **Adafruit_IS31FL3731** folder contains **Adafruit_IS31FL3731.cpp** and **Adafruit_IS31FL3731.h**

Place the **Adafruit_IS31FL3731** library folder your **arduinorsketchfolder/libraries/** folder. You may need to create the **libraries** subfolder if its your first library. Restart the IDE.

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 \(http://adafru.it/aJa\)](http://adafru.it/aJa)

[Download Adafruit_GFX Library](http://adafru.it/cBB)

<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 **arduinorsketchfolder/libraries/** folder like you did with the IS31FL library

Load Demo