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.

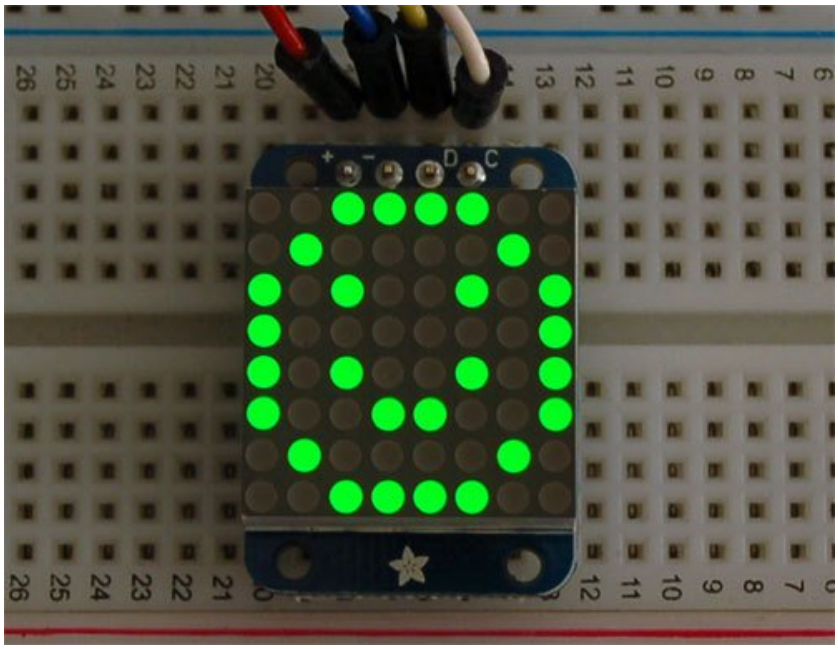
We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

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

Adafruit LED Backpacks

Created by lady ada



Last updated on 2016-12-30 04:06:36 PM UTC

Guide Contents

Guide Contents 2
Overview 4
1.2" 8x8 Matrix 8
(http://adafru.it/aPT)Mini 8x8 Matrix Software 12
0.8" 8x8 Matrix 15
Mini 8x8 Matrix Software 19
0.54" Alphanumeric 22
Attaching the Backpack 22
Attaching Header 27
Prepare the header strip: 28
Add the Backpack: 28
Downloading the Arduino Library 29
Wiring! 29
Load Demo 30
Library Reference 32
ASCII data 33
Writing Data 34
0.56" 7-Segment Backpack 35
Seven-Segment Backpack Firmware 38
1.2" 7-segment Backpack 42
Arduino Wiring - R3 and later 47
Arduino Due and Other 3.3v Processors 47
Arduino "Classic" Wiring 48
Seven-Segment Backpack Firmware 48
Bi-Color 8x8 Matrix 51
Bi-Color 8x8 LED Backpack Firmware 55
Schematic 57
Bi-Color 24 Bargraph 58
Attaching the bar-graph modules 59
Soldering on breadboard pins 63
Bi-Color Bargraph LED Backpack Wiring & Firmware 65
Connecting Multiple Backpacks 69

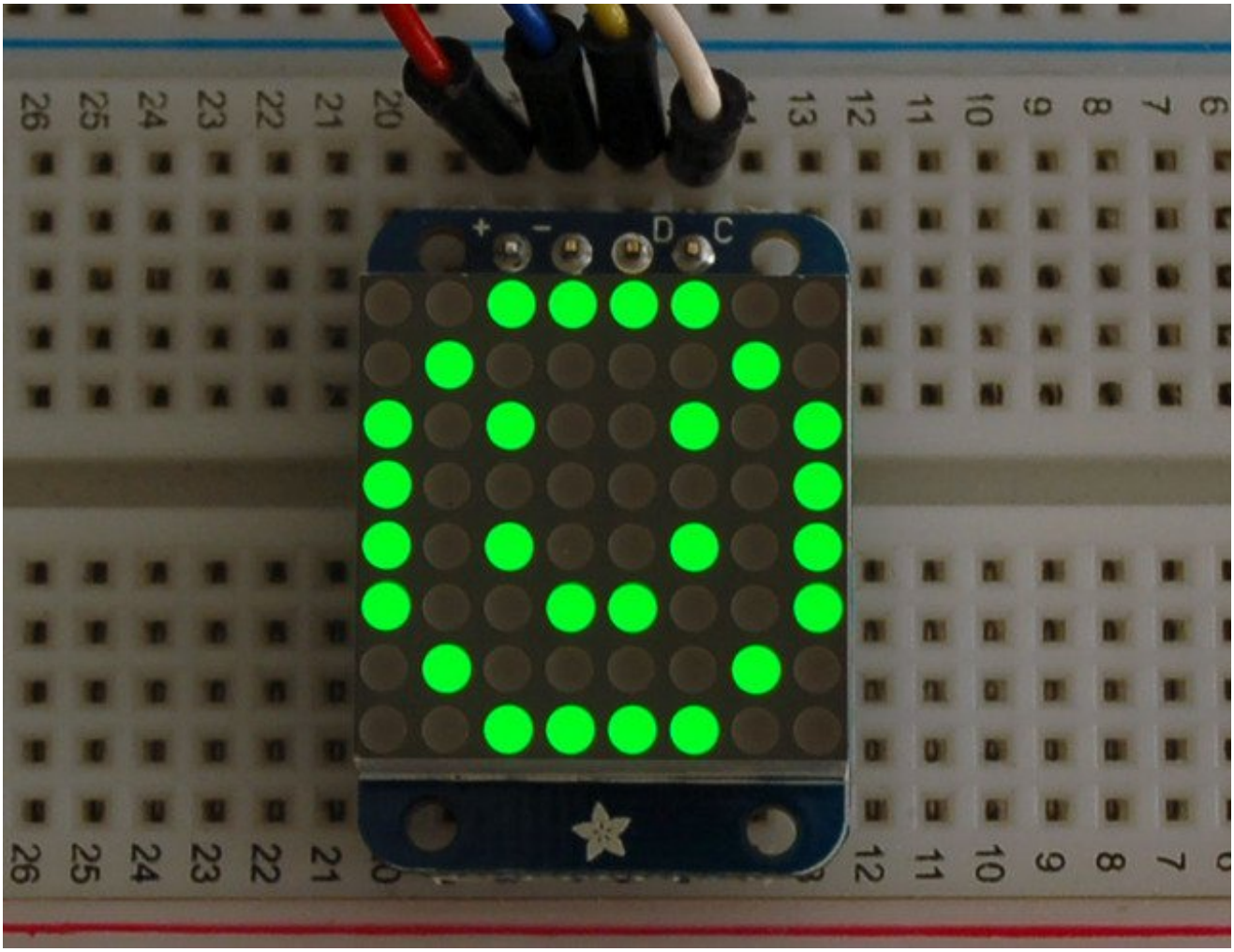
Wire it Up 70
Configure the Address 70
Changing I2C Address 72
Changing Addresses 72
Changing the address in your code 73
F.A.Q. 75
Downloads 76
Software 76
Files 76
HT16K33 8x16 LED Backpack Breakout 76
8x8 0.8" LED Backpack 78
8x8 1.2" LED Backpack 80
8x8 1.2" Bi-Color LED Backpack 82
16x8 1.2" LED Backpacks 83
Quad 0.56" 7-Segment 84
Quad 0.54" 14-segment Alphanumeric 85
Quad 1.2" 7-Segment 86
Bicolor 24-Bargraph 87

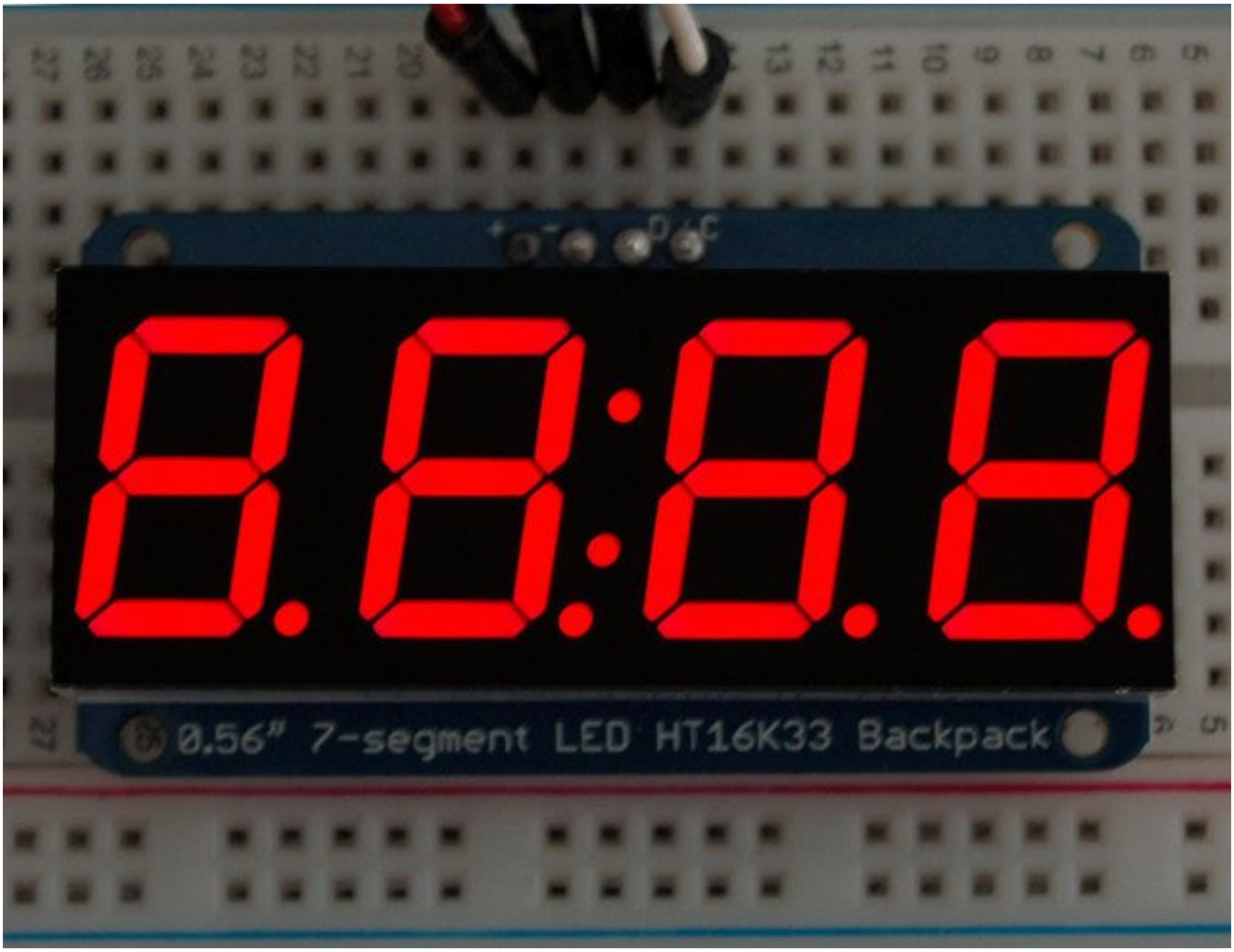
Overview

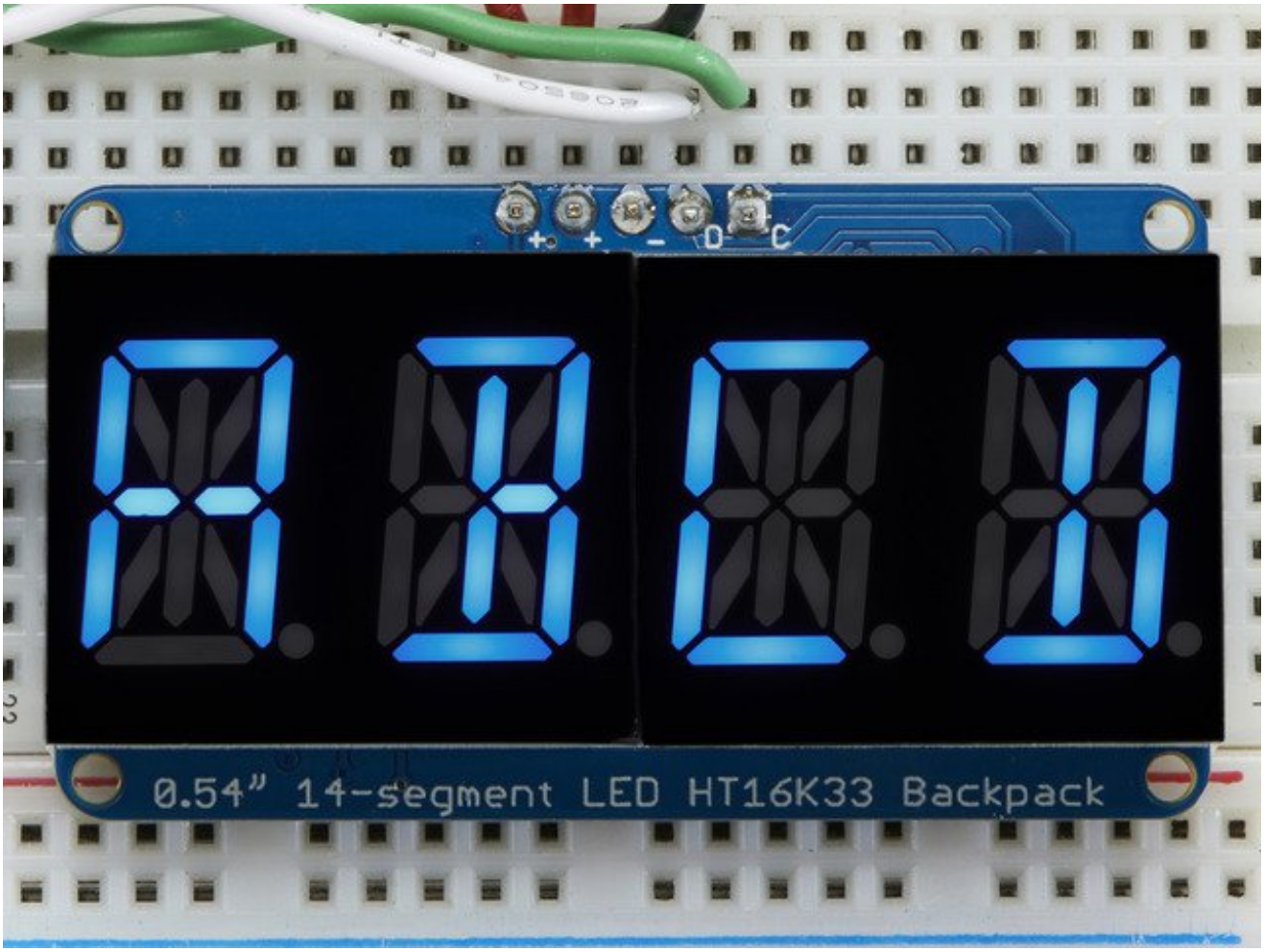
What's better than a single LED? Lots of LEDs! A fun way to make a small display is to use an 8x8 matrix (http://adafru.it/aLG) or a 4-digit 7-segment display (http://adafru.it/aLH). Matrices like these are 'multiplexed' - so to control 64 LEDs you need 16 pins. That's a lot of pins, and there are driver 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. Here at Adafruit we feel your pain! After all, wouldn't it be awesome if you could control a matrix without tons of wiring? That's where these adorable LED matrix backpacks come in.

We have them in quite a few flavors!

- Adorable Mini 8x8 (http://adafru.it/ttf)
- Classic 1.2" 8x8 (round and square dots) (http://adafru.it/ttA)
- 4-digit 0.56" 7-segment (http://adafru.it/ttB)
- 4-digit 1.2" 7-segment (http://adafru.it/ttC)
- 4-digit 0.54" 14-segment Alphanumeric (http://adafru.it/ttD)
- Bi-color 8x8 (http://adafru.it/902)
- Bi-color Bargraph (http://adafru.it/1721)


+
-
D
C


8.8:8.8.
0.56" 7-segment LED HT16K33 Backpack



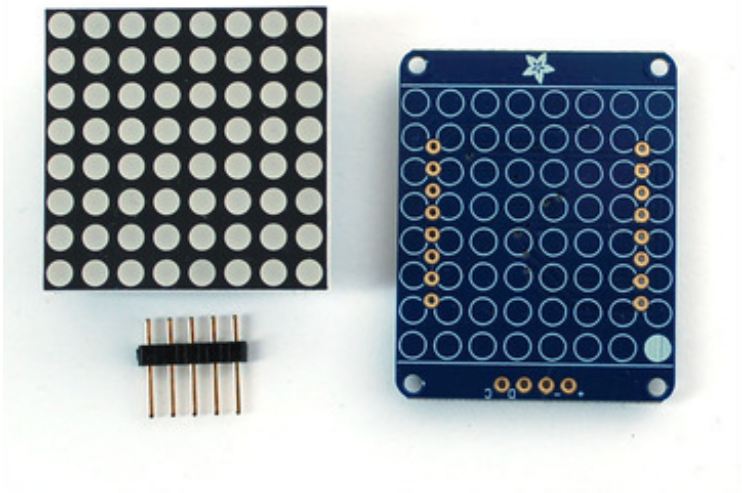
The matrices use a driver chip that does all the heavy lifting for you: They have a built in clock so they multiplex the display. They use constant-current drivers for ultra-bright, consistant color (the images above are photographed at the dimmest setting to avoid overloading our camera!), 1/16 step display dimming, all via a simple I2C interface. The backpacks come with address-selection jumpers so you can connect up to four mini 8x8's or eight 7-segments (or a combination, such as four mini 8x8's and four 7-segments, etc) on a single I2C bus.

The product kit comes with a fully tested and assembled LED backpack, a 4-pin header and the matrix of your choice. A bit of soldering is required to attach the matrix onto the backpack 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. If you've been eyeing matrix displays but hesitated because of the complexity, his is the solution you've been looking for!

1.2" 8x8 Matrix

This version of the LED backpack is designed for the 1.2" 8x8 matrices. They measure only 1.2"x1.2" so its a shame to use a massive array of chips to control it. This backpack solves the annoyance of using 16 pins or a bunch of chips by having an I2C constant-current matrix controller sit neatly on the back of the PCB. The controller chip takes care of everything, drawing all 64 LEDs in the background. All you have to do is write data to it using the 2-pin I2C interface. There are two address select pins so you can select one of 8 addresses to control up to 8 of these on a single 2-pin I2C bus (as well as whatever other I2C chips or sensors you like). The driver chip can 'dim' the entire display from 1/16 brightness up to full brightness in 1/16th steps. It cannot dim individual LEDs, only the entire display at once.
These instruction apply to the 1.2" Matrix only! If you have a Bi-Color or 0.8" square matrix, follow the links on the left side of the page.



When you buy a pack from Adafruit, it comes with the fully tested and assembled backpack as well as a 8x8 matrix in one of the colors we provide (say, red, yellow or green). You'll need to solder the matrix onto the backpack but its an easy task.

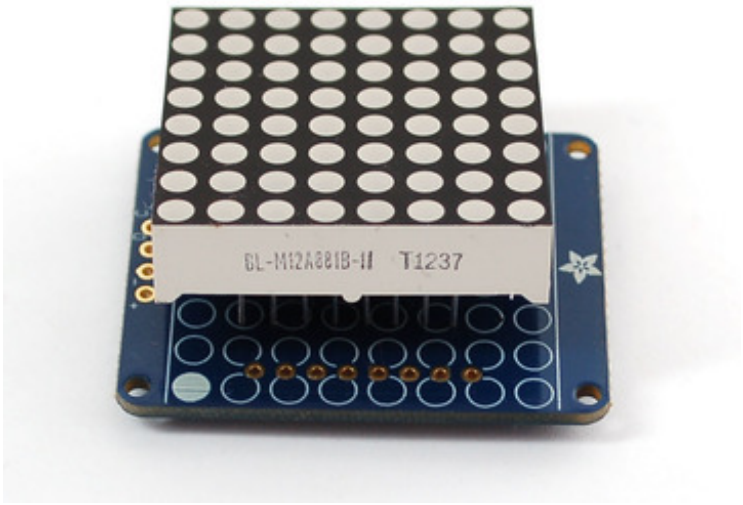
-

WATCH OUT! THE MATRIX

-

MUST BE INSTALLED THE RIGHT WAY!

First look for the line of text on the side of the LED matrix

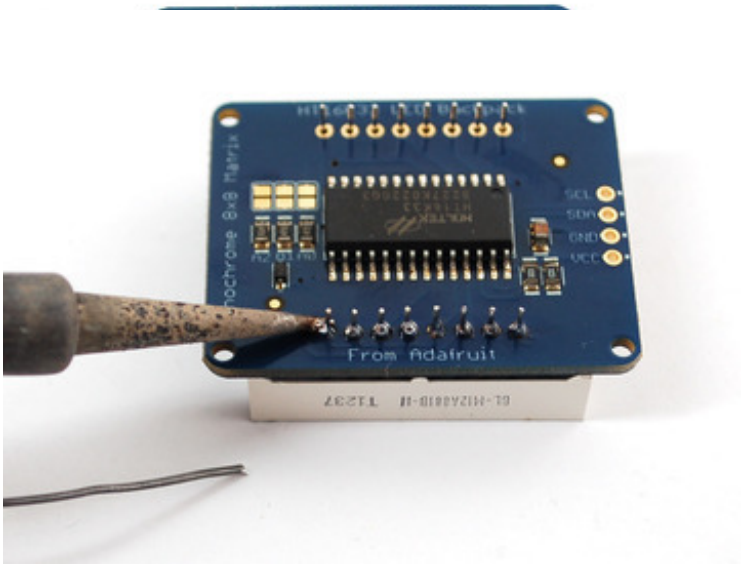
- 

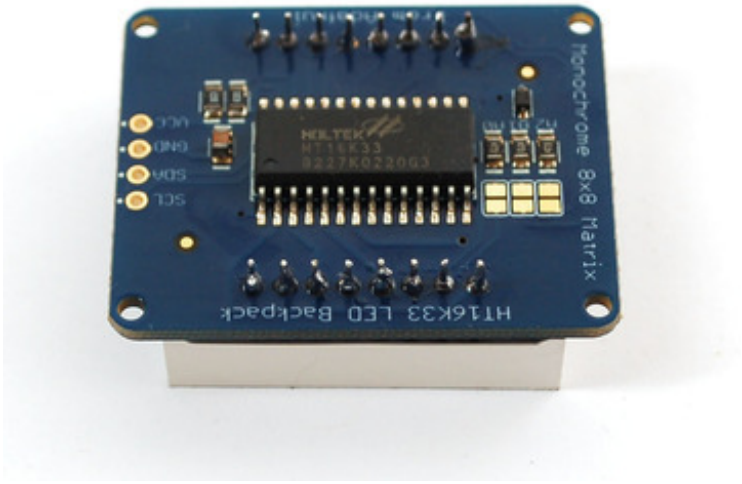
WATCH OUT! THE MATRIX MUST BE INSTALLED THE RIGHT WAY!

Find the corner of the backpack with a filled in dot. Make sure that the text on the side of the matrix is on the same side as the filled dot

WATCH OUT! THE MATRIX MUST BE INSTALLED THE RIGHT WAY!

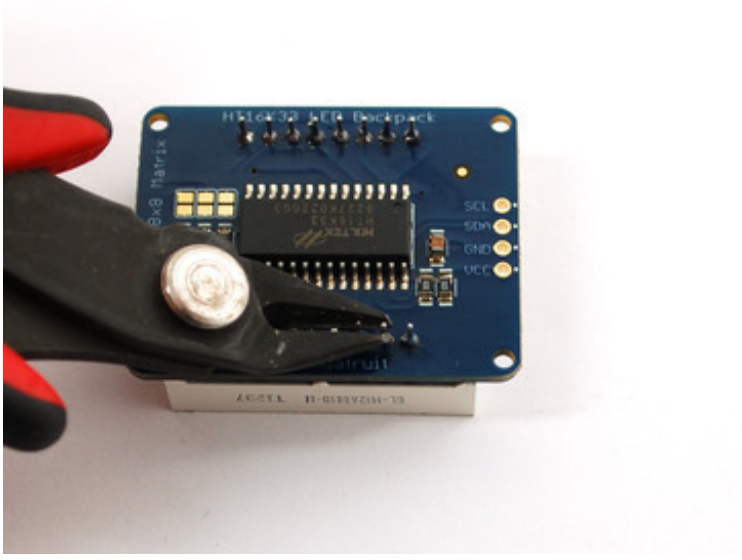
Slide the matrix into the backpack and flip it over. Triple check that the text is on the same side as the **From Adafruit** text

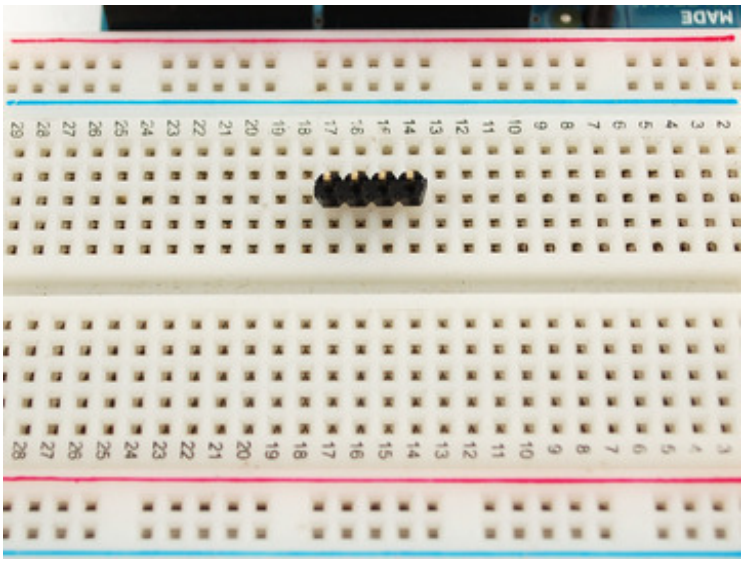


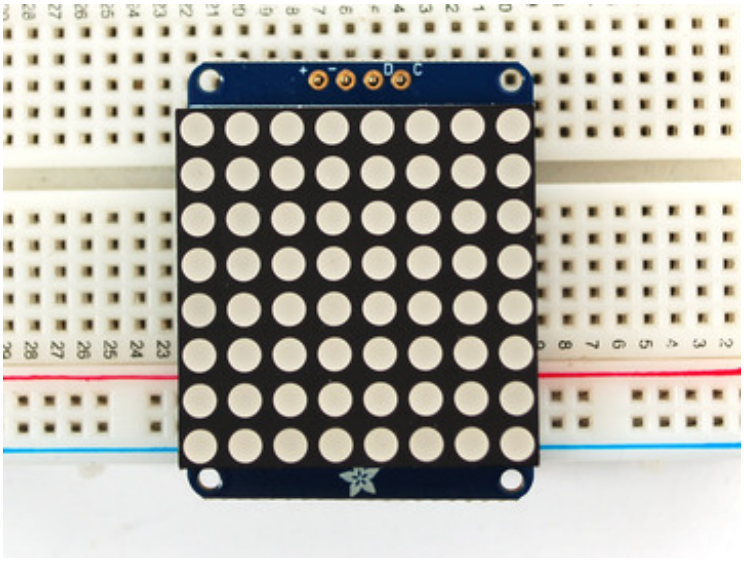


Solder in all 16 pins

Then clip the matrix leads short

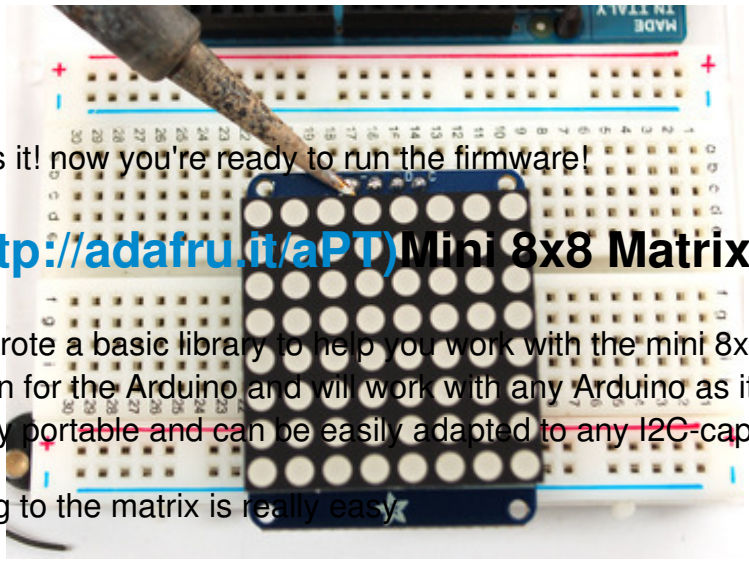


- 
  Now you're ready to wire it up to a microcontroller. We'll assume you want to use a 4pin header. You can also of course solder wires directly. Place a 4-pin piece of header with the LONG pins down into the breadboard.



Place the soldered backpack on top of the header.

Solder the four pins

- 

That's it! now you're ready to run the firmware!

(http://adafru.it/aPT)Mini 8x8 Matrix Software

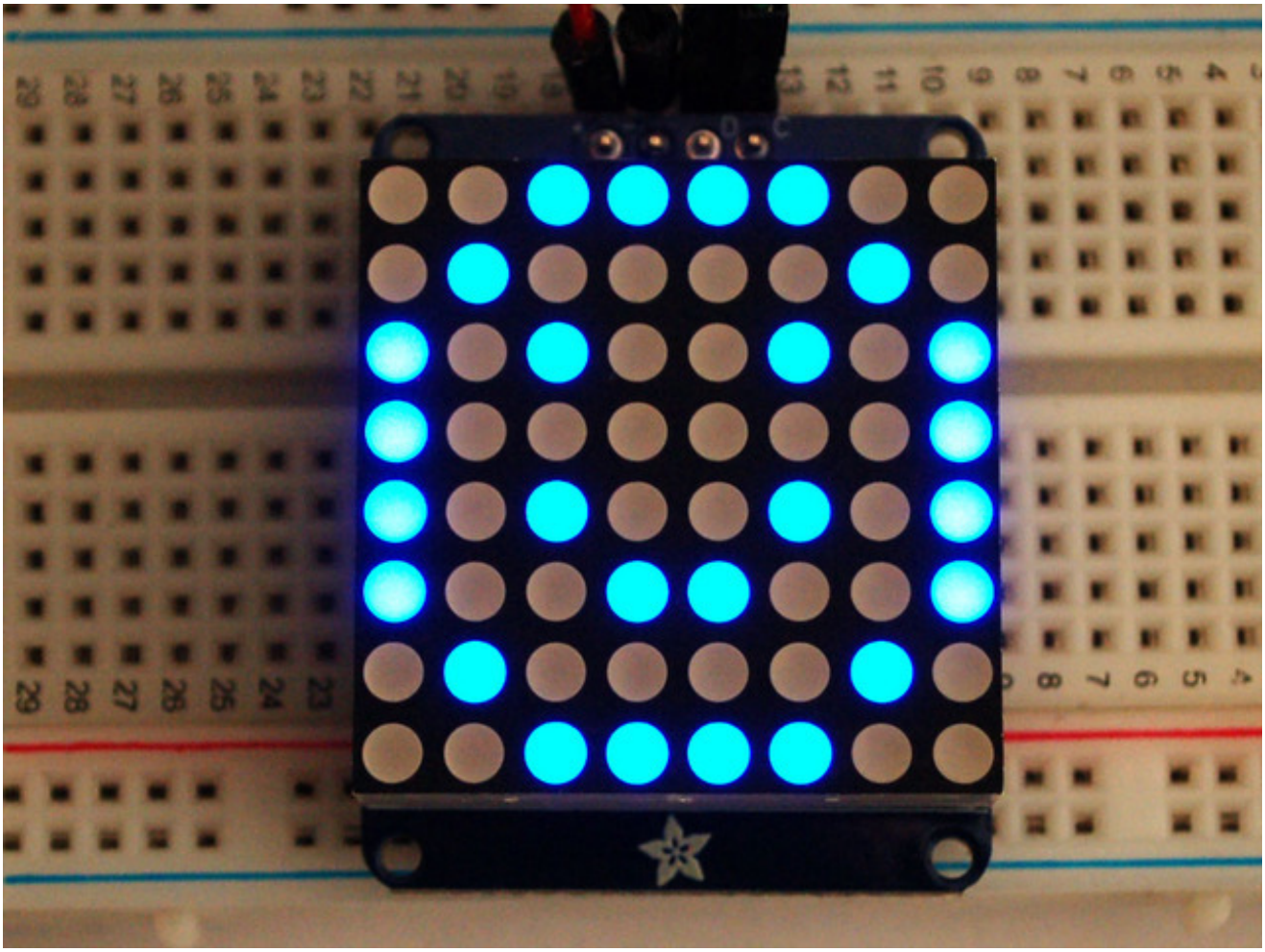
We wrote a basic library to help you work with the mini 8x8 matrix backpack. The library is written for the Arduino and will work with any Arduino as it just uses the I2C pins. The code is very portable and can be easily adapted to any I2C-capable micro.

Wiring to the matrix is really easy

-
- Connect **CLK** to the I2C clock - on Arduino UNO thats Analog #5, on the Leonardo its Digital #3, on the Mega its digital #21
- Connect **DAT** to the I2C data - on Arduino UNO thats Analog #4, on the Leonardo its Digital #2, on the Mega its digital #20
- Connect **GND** to common ground
- Connect **VCC+** to power - 5V is best but 3V also seems to work for 3V microcontrollers.

Next, download the Adafruit LED Backpack library from github (http://adafru.it/aLI) . To download click the DOWNLOADS button in the top right corner, 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 its your first library. You'll also need to download the Adafruit GFX library (http://adafru.it/aJa) that provides the graphics drawing routines. Restart the IDE.

Once you've restarted you should be able to select the**File->Examples->Adafruit_LEDBackpack->matrix88** example sketch. Upload it to your Arduino as usual. You should see a basic test program that goes through a bunch of different drawing routine



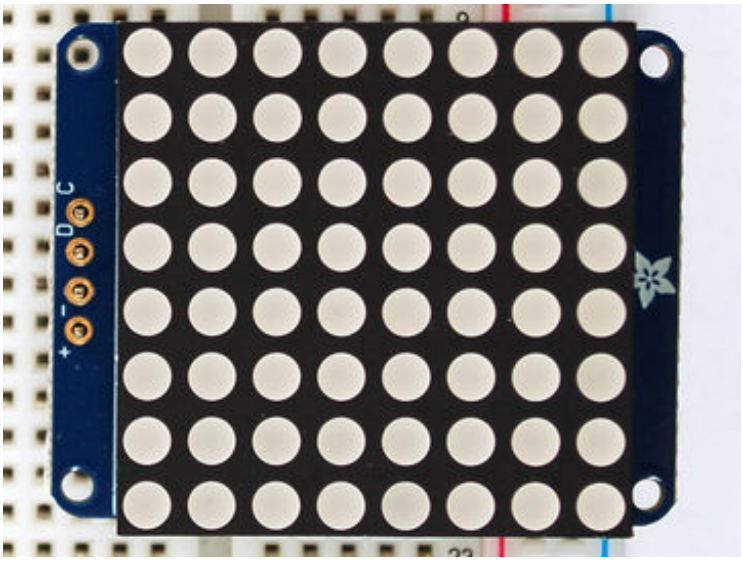
Once you're happy that the matrix works, you can write your own sketches. The 8x8 matrix supports everything the Adafruit GFX library - drawing pixels, lines, rectangles, circles, triangles, roundrects, and small bitmaps. For more details check out the GFX page which will detail all of the GFX routines (http://adafru.it/aPx).

All the drawing routines only change the display memory kept by the Arduino. Don't forget to call **writeDisplay()** after drawing to 'save' the memory out to the matrix via I2C.

There are also a few small routines that are special to the matrix:

- **setBrightness(*brighness*)**- will let you change the overall brightness of the entire display. 0 is least bright, 15 is brightest and is what is initialized by the display when you start. You can call this function at any time to change the brightness of the -entire- display

- **blinkRate(*rate*)** - You can blink the entire display. 0 is no blinking. 1, 2 or 3 is for display blinking.You can call this function at any time to change the blink rate of the -entire- display

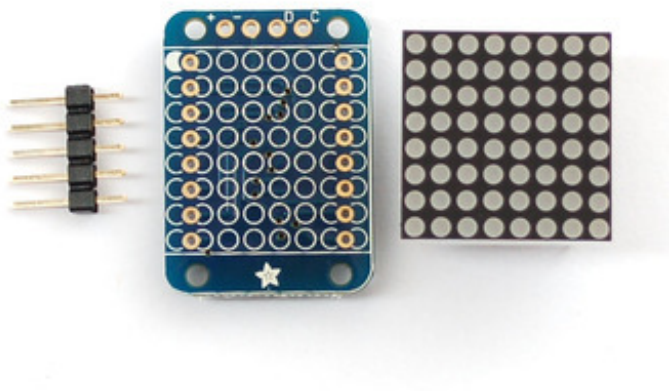
- 

The default orientation for graphics commands on this display places pixel (0,0) at the top-left when the header is at the left and Adafruit logo at the right. To use the matrix as shown above (header at top, logo at bottom), call matrix.setRotation(3) before issuing graphics commands.

0.8" 8x8 Matrix

This version of the LED backpack is designed for these very cute miniature 8x8 matrices. They measure only 0.8"x0.8" so its a shame to use a massive array of chips to control it. This backpack solves the annoyance of using 16 pins or a bunch of chips by having an I2C constant-current matrix controller sit neatly on the back of the PCB. The controller chip takes care of everything, drawing all 64 LEDs in the background. All you have to do is write data to it using the 2-pin I2C interface. There are two address select pins so you can select one of 4 addresses to control up to 4 of these on a single 2-pin I2C bus (as well as whatever other I2C chips or sensors you like). The driver chip can 'dim' the entire display from 1/16 brightness up to full brightness in 1/16th steps. It cannot dim individual LEDs, only the entire display at once.

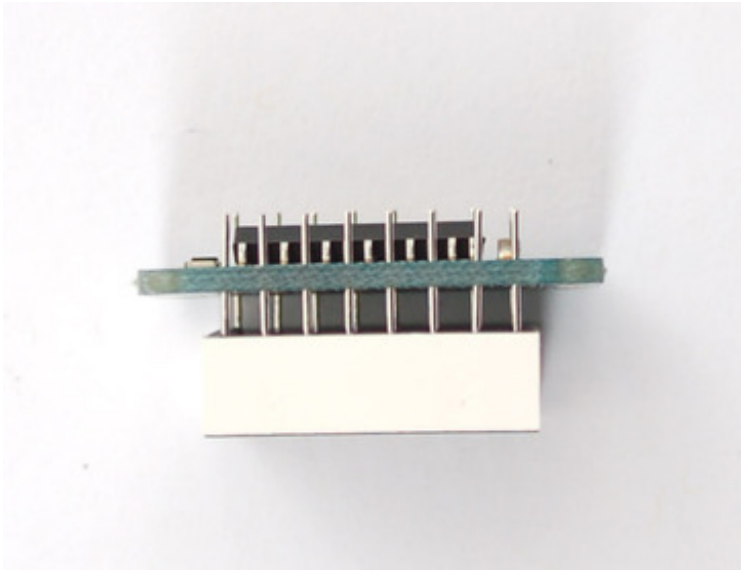
These instruction apply to the 0.8" Matrix only! If you have a Bi-Color or 1.2" square matrix, follow the links on the left side of the page.



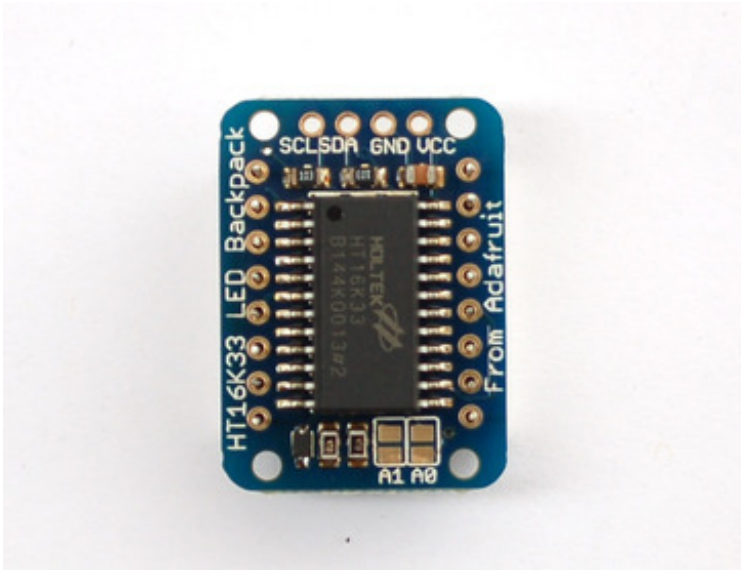
When you buy a pack from Adafruit, it comes with the fully tested and assembled backpack as well as a 8x8 matrix in one of the colors we provide (say, red, yellow or green). You'll need to solder the matrix onto the backpack but its an easy task.

-

Remove the parts from packaging and place the LED

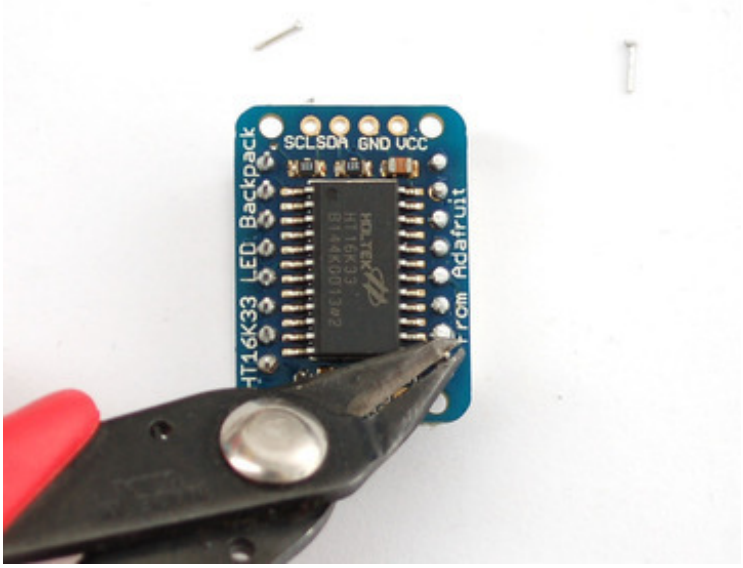
- 

matrix OVER the silkscreen side. It can go 'either way' - the matrix is symmetric so as long as it goes onto the front it will work in any orientation. Do not solder the matrix over the chip on the back of the backpack - it will not work then!

- 

Turn the backpack over so its sitting flat on the matrix.

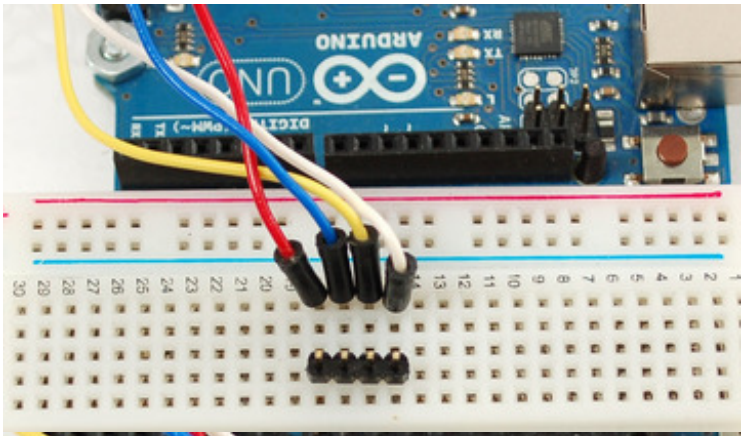
Solder all 16 pins.

- 

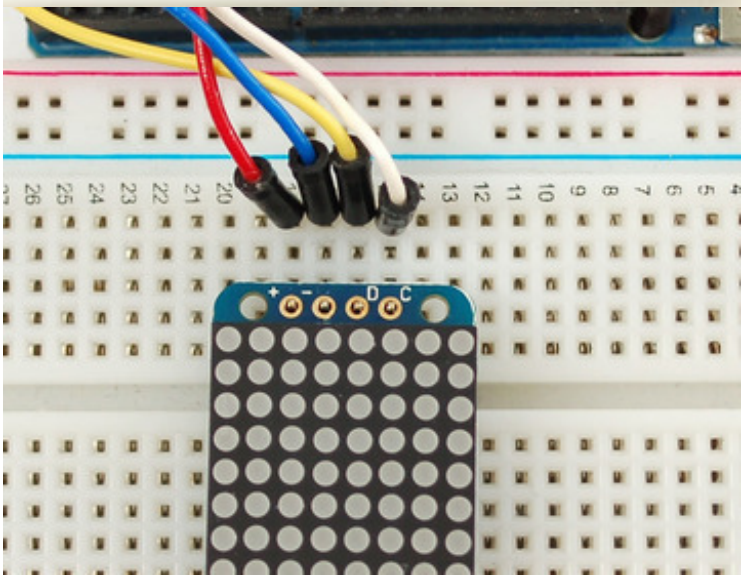
-

Clip the long pins.

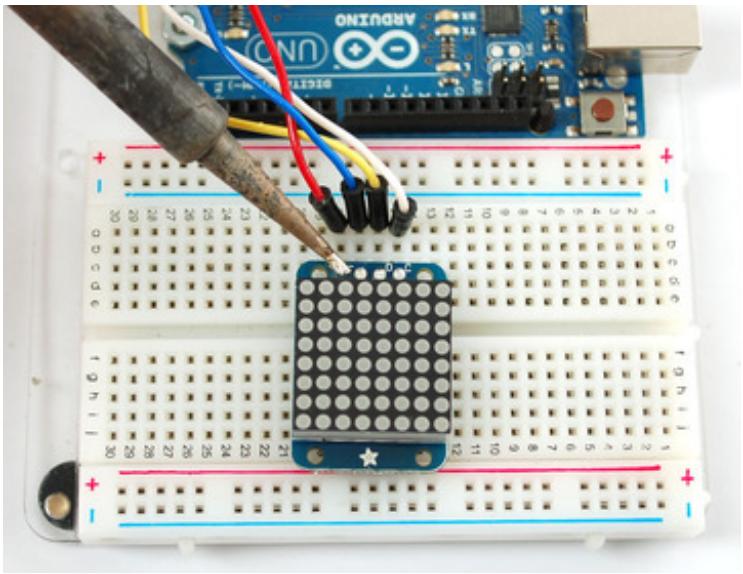
Now you're ready to wire it up to a microcontroller. We'll assume



you want to use a 4pin header. You can also of course solder wires directly. Place a 4-pin piece of header with the LONG pins down into the breadboard.



Place the soldered backpack on top of the header.



Solder 'em!

That's it! now you're ready to run the firmware!

Mini 8x8 Matrix Software

We wrote a basic library to help you work with the mini 8x8 matrix backpack. The library is written for the Arduino and will work with any Arduino as it just uses the I2C pins. The code is very portable and can be easily adapted to any I2C-capable micro.

Wiring to the matrix is really easy

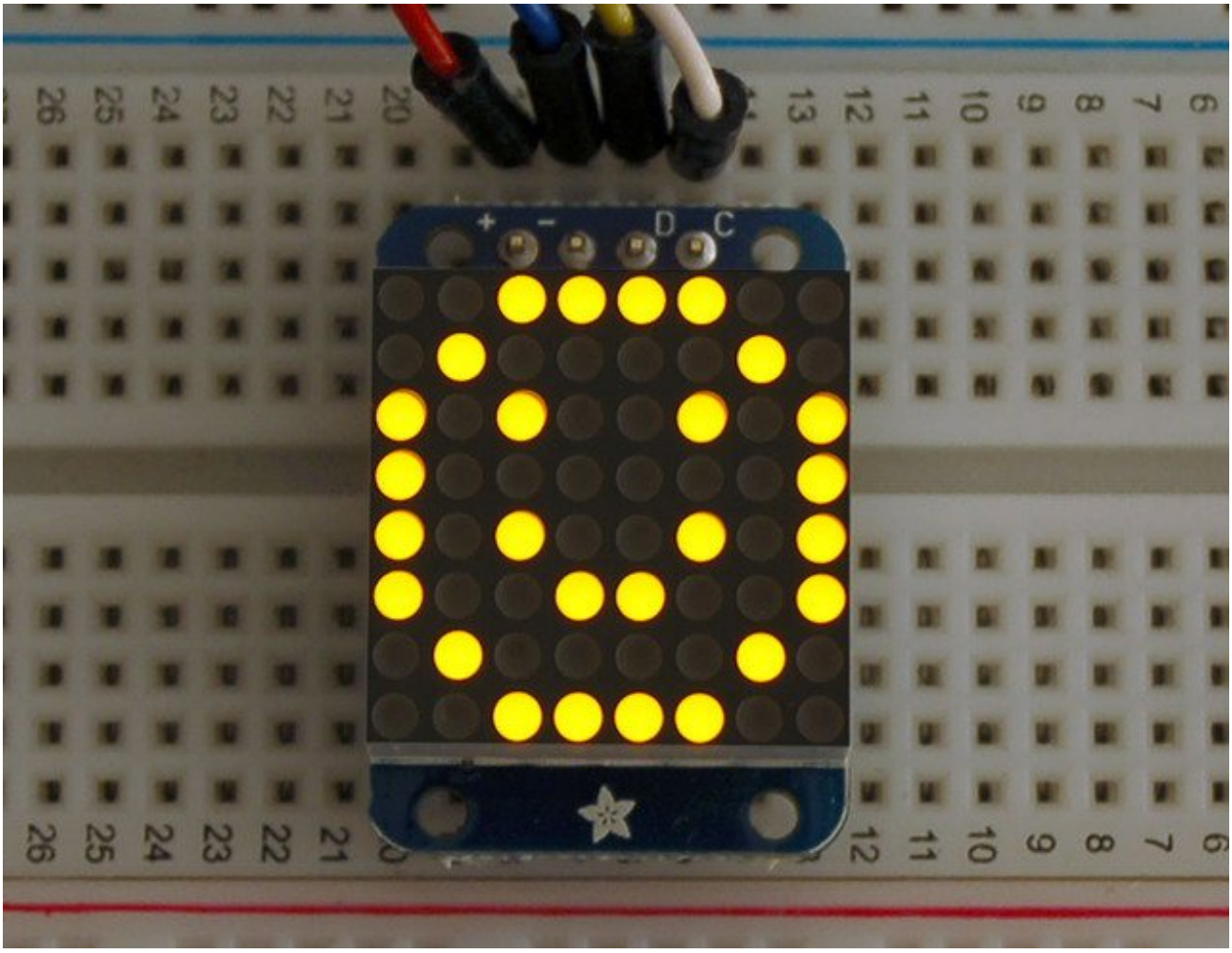
- Connect **CLK** to the I2C clock - on Arduino UNO thats Analog #5, on the Leonardo its Digital #3, on the Mega its digital #21
- Connect **DAT** to the I2C data - on Arduino UNO thats Analog #4, on the Leonardo its Digital #2, on the Mega its digital #20
- Connect **GND** to common ground
- Connect **VCC+** to power - 5V is best but 3V also seems to work for 3V microcontrollers.

Next, download the Adafruit LED Backpack library from github (http://adafru.it/aLI) . To download click the DOWNLOADS button in the top right corner, 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 its your first library.

You'll also need to download the Adafruit GFX library (http://adafru.it/aJa) that provides the graphics drawing routines. Follow the same instructions as above, but with Adafruit_GFX instead of Adafruit_LEDBackpack.

Restart the IDE.

Once you've restarted you should be able to select the**File->Examples->Adafruit_LEDBackpack->matrix88** example sketch. Upload it to your Arduino as usual. You should see a basic test program that goes through a bunch of different drawing routines

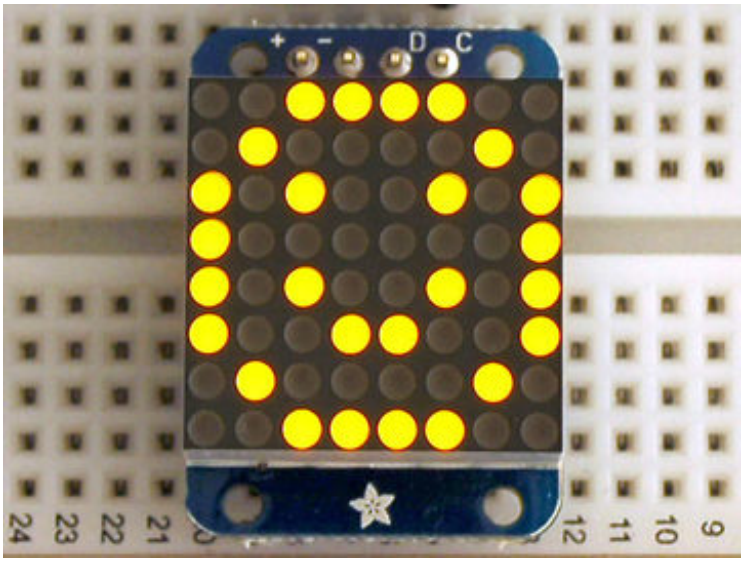


Once you're happy that the matrix works, you can write your own sketches. The 8x8 matrix supports everything the Adafruit GFX library - drawing pixels, lines, rectangles, circles, triangles, roundrects, and small bitmaps. [For more details check out the GFX page which will detail all of the GFX routines](http://adafru.it/aPx) (http://adafru.it/aPx).

All the drawing routines only change the display memory kept by the Arduino. Don't forget to call **writeDisplay()** after drawing to 'save' the memory out to the matrix via I2C.

There are also a few small routines that are special to the matrix:

- **setBrightness(*brighness*)**- will let you change the overall brightness of the entire display. 0 is least bright, 15 is brightest and is what is initialized by the display when you start
- **blinkRate(*rate*)** - You can blink the entire display. 0 is no blinking. 1, 2 or 3 is for display blinking.

- 

The default orientation for graphics commands on this display places pixel (0,0) at the top-left when the header is at the top and Adafruit logo at the bottom. matrix.setRotation() can be used to use the display in other orientations.

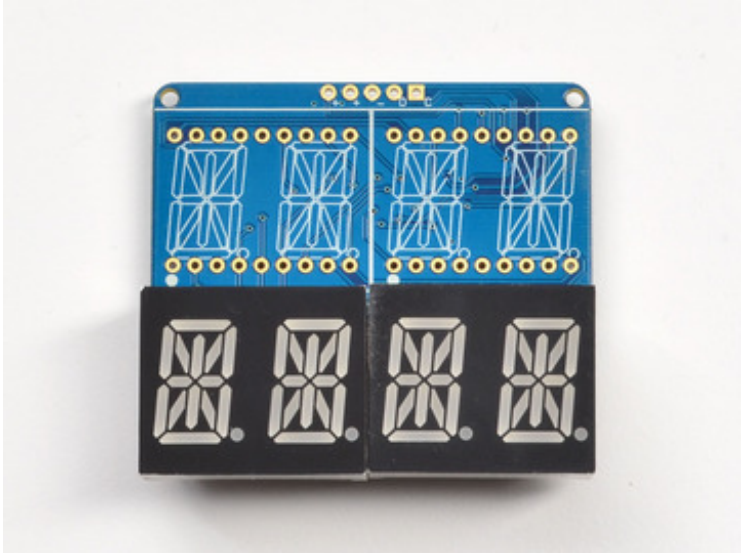
0.54" Alphanumeric

(http://adafru.it/l3c)

This version of the LED backpack is designed for two dual 14-segment "Alphanumeric" displays. These 14-segment displays normally require 18 pins (4 'characters' and 14 total segments each) This backpack solves the annoyance of using 18 pins or a bunch of chips by having an I2C constant-current matrix controller sit neatly on the back of the PCB. The controller chip takes care of everything, drawing all the LEDs in the background. All you have to do is write data to it using the 2-pin I2C interface.

There are three address select pins so you can select one of 8 addresses to control up to 8 of these on a single 2-pin I2C bus (as well as whatever other I2C chips or sensors you like). The driver chip can 'dim' the entire display from 1/16 brightness up to full brightness in 1/16th steps. It cannot dim individual LEDs, only the entire display at once.

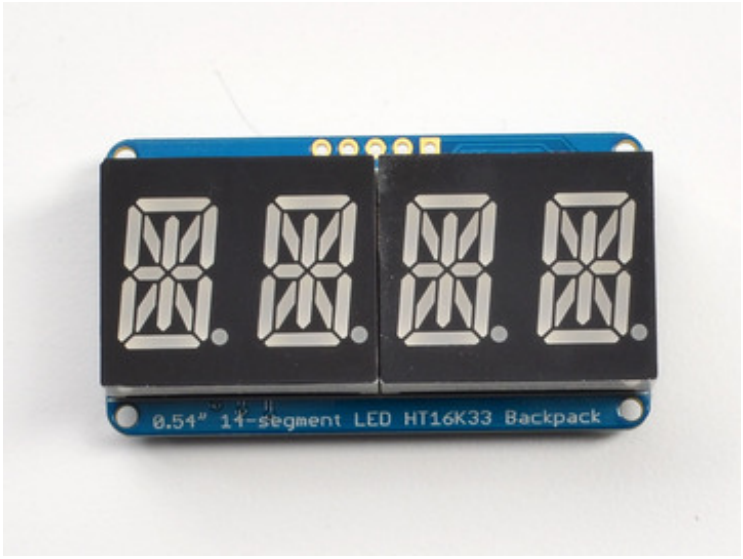
Attaching the Backpack

- 

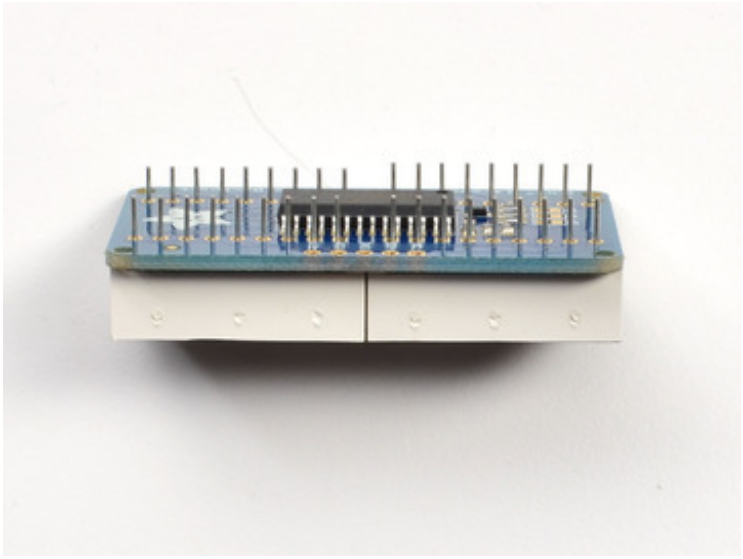
When you buy a pack from Adafruit, it comes with the fully tested and assembled backpack as well as two dual 14-segment display in one of the colors we provide (say, red, yellow, blue or green). You'll need to solder the matrix onto the backpack but it's an easy task.

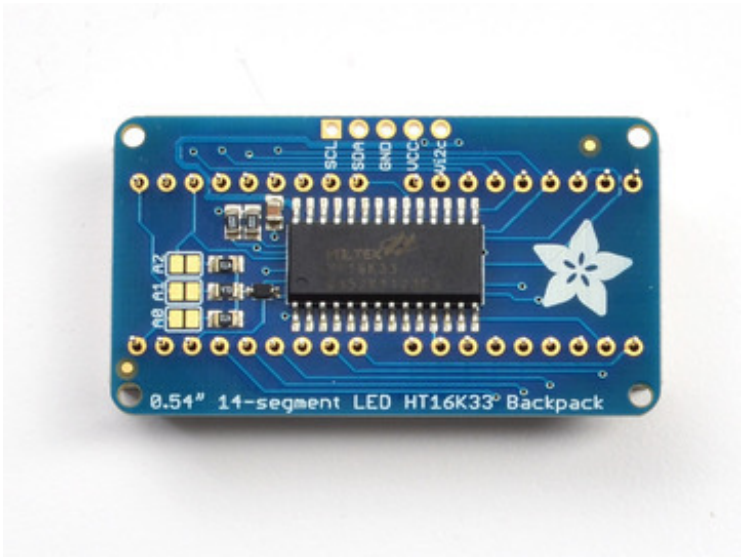
Remove the parts from packaging and place the LED matrices OVER the silkscreen side. **DO NOT PUT THE DISPLAY ON UPSIDE DOWN**

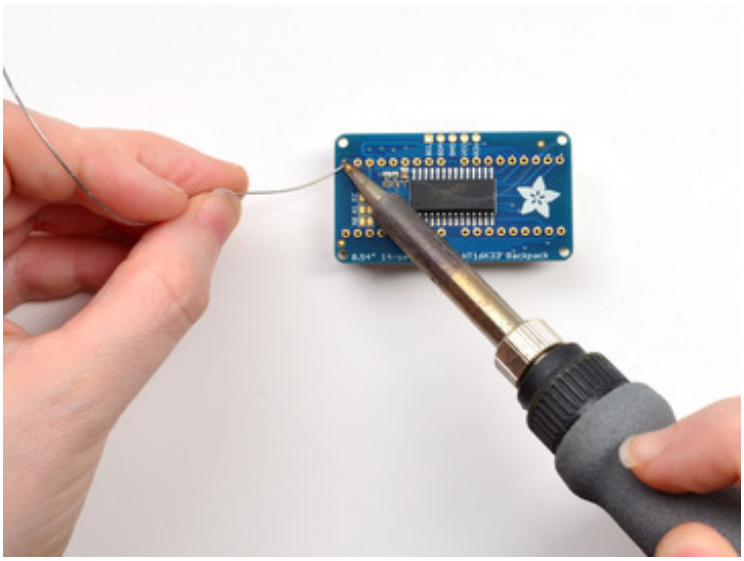
OR IT WONT WORK!! Check the image below to make sure the 'decimal point' dots are on the bottom, matching the silkscreen.
Remove the parts from packaging and place the LED matrices OVER the silkscreen side. **DO NOT PUT THE DISPLAY ON UPSIDE DOWN OR IT WONT WORK!! Check the image below to make sure the 'decimal point' dots are on the bottom, matching the silkscreen.** Remove the parts from packaging and place the LED matrices OVER the silkscreen side. **DO NOT PUT THE DISPLAY ON UPSIDE DOWN OR IT WONT WORK!! Check the image below to make sure the 'decimal point' dots are on the bottom, matching the silkscreen.**

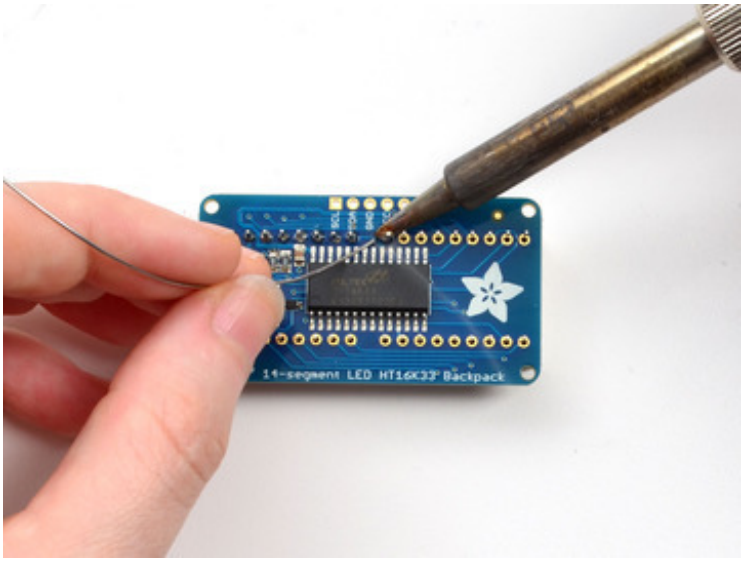
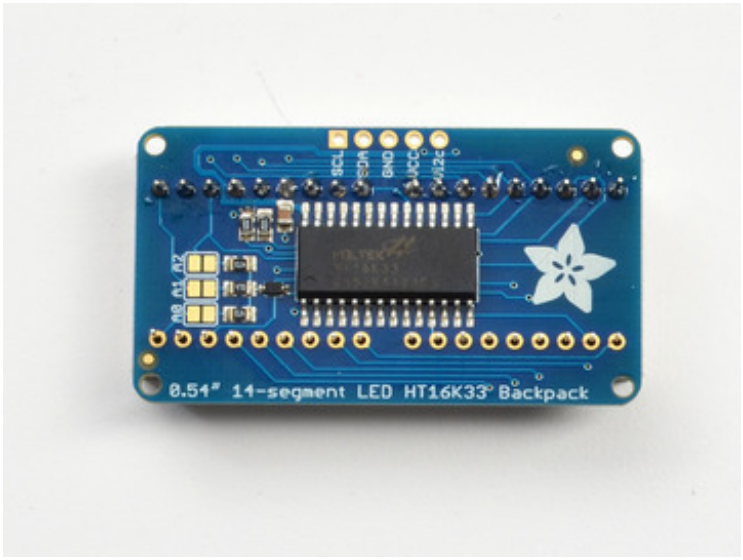
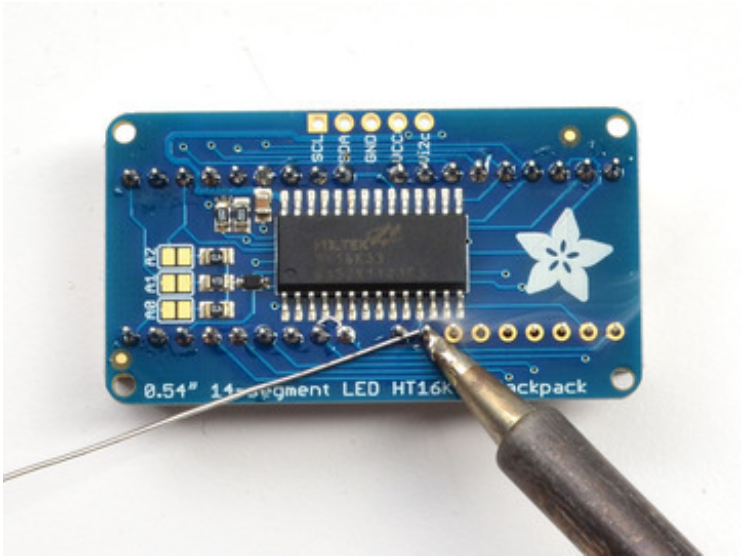
- 

Remove the parts from packaging and place the LED matrices OVER the silkscreen side. **DO NOT PUT THE DISPLAY ON UPSIDE DOWN OR IT WONT WORK!! Check the image below to make sure the 'decimal point' dots are on the bottom, matching the silkscreen.**

- 

  Turn the backpack over so it is sitting flat on the matrix.
- 

- 

- 
- 

Solder all of the pins!
-