



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

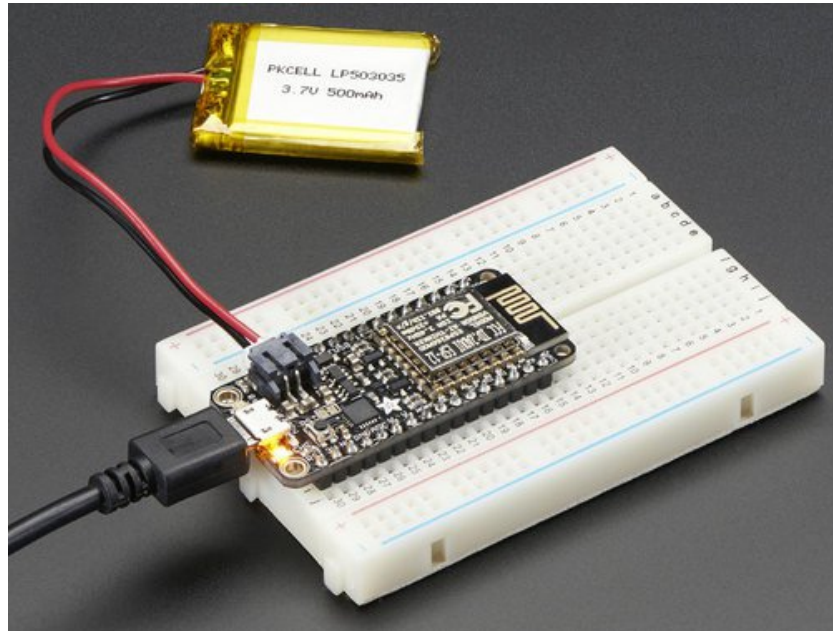
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



□

Adafruit Feather HUZDAH ESP8266

Created by lady ada



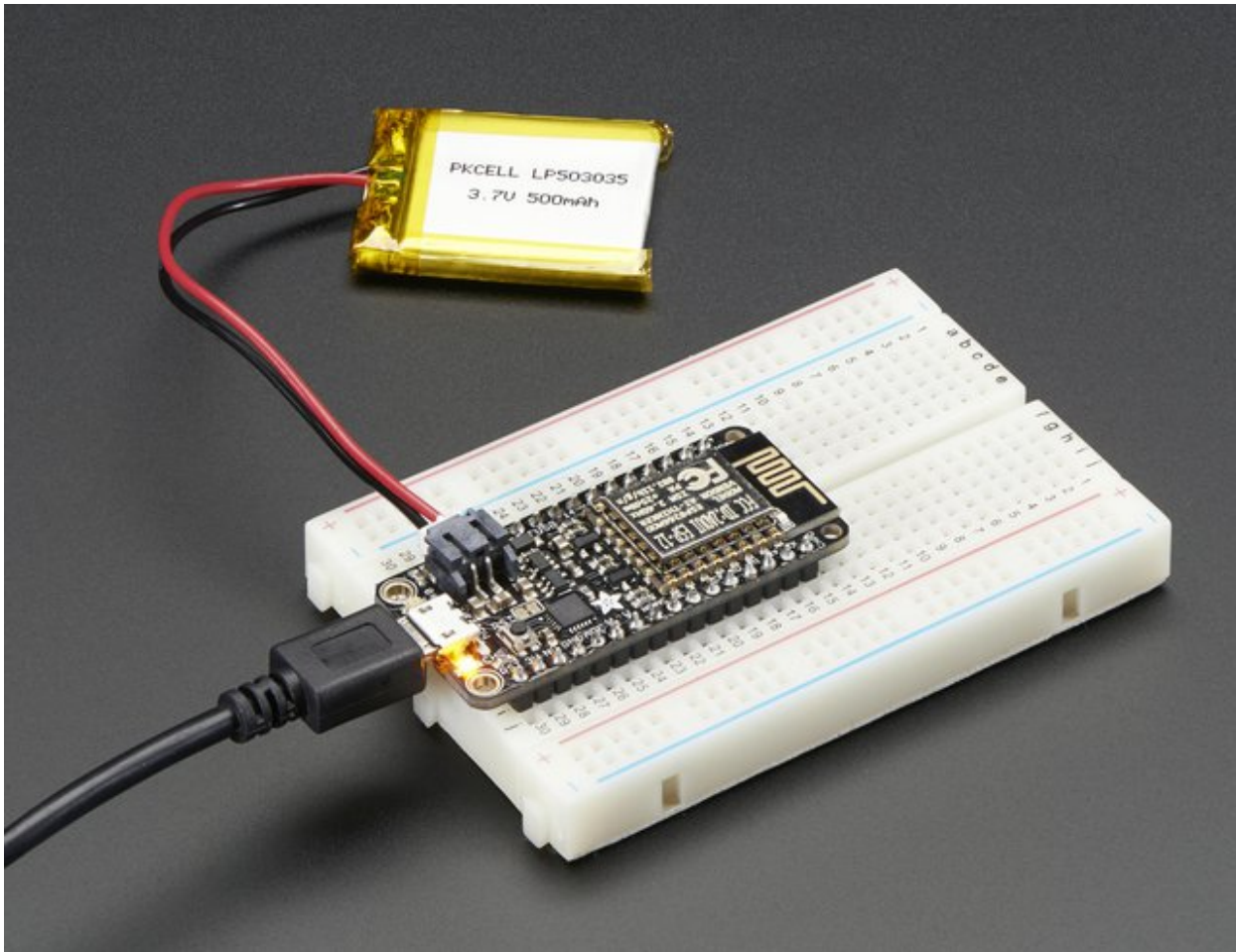
Last updated on 2017-04-27 07:27:44 PM UTC

Guide Contents

Guide Contents	2
Overview	4
Pinouts	9
Power Pins	10
Logic pins	11
Serial pins	11
I2C & SPI pins	11
GPIO pins	12
Analog Pins	13
Other control pins	14
NC Pins	15
Assembly	16
Header Options!	16
Soldering in Plain Headers	19
Prepare the header strip:	20
Add the breakout board:	21
And Solder!	21
Soldering on Female Header	24
Tape In Place	24
Flip & Tack Solder	25
And Solder!	26
Power Management	29
Battery + USB Power	29
Power supplies	30
Measuring Battery	31
ENable pin	31
Using NodeMCU Lua	33
Open up serial console	33
Hello world!	35
Scanning & Connecting to WiFi	36
WebClient example	37
Using Arduino IDE	38

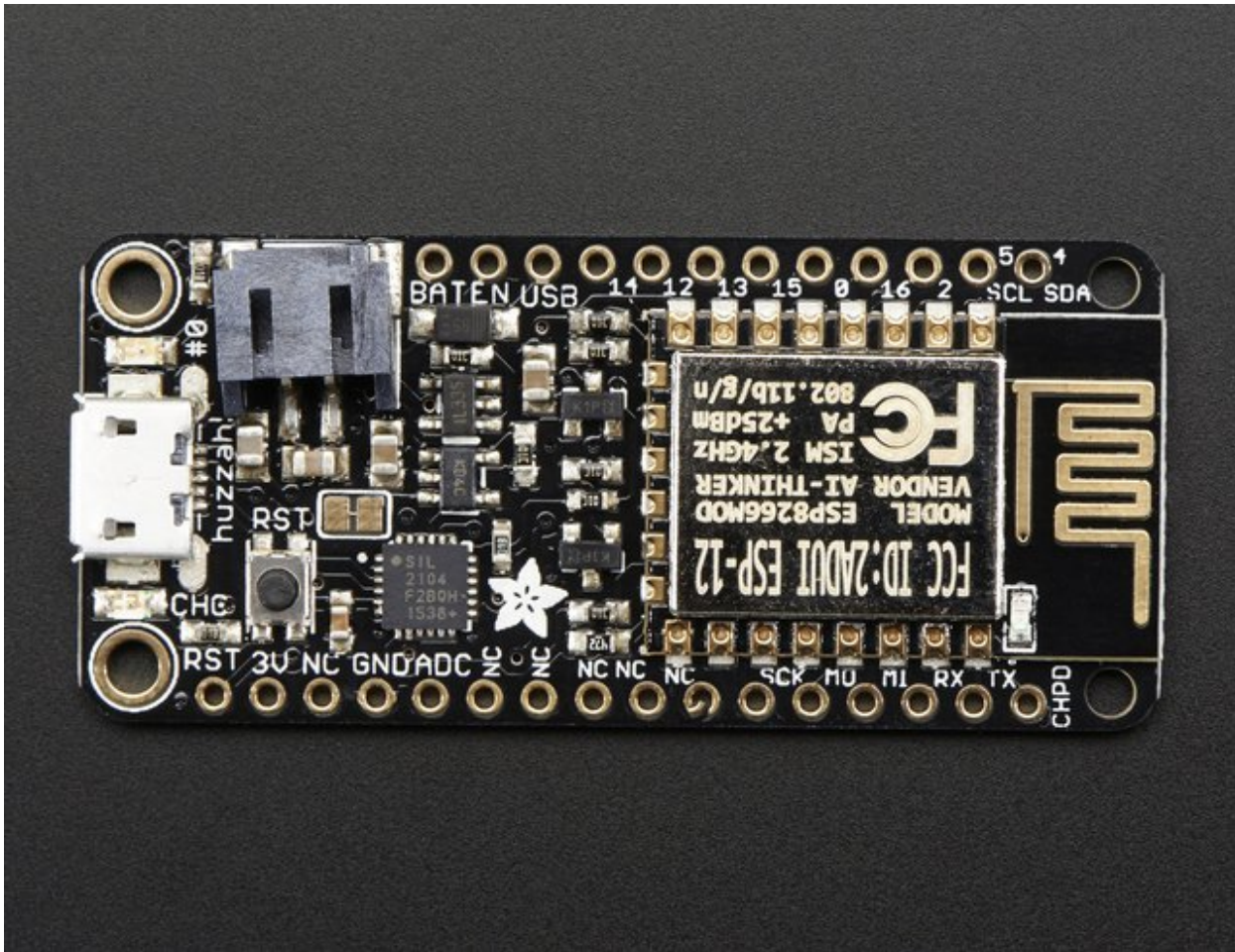
Install the Arduino IDE 1.6.8 or greater	39
Install the ESP8266 Board Package	39
Setup ESP8266 Support	40
Blink Test	42
Connecting via WiFi	43
Downloads	47
Datasheets & Files	47
More info about the ESP8266	47
Schematic	47
Fabrication Print	48
ESP8266 F.A.Q.	50

Overview

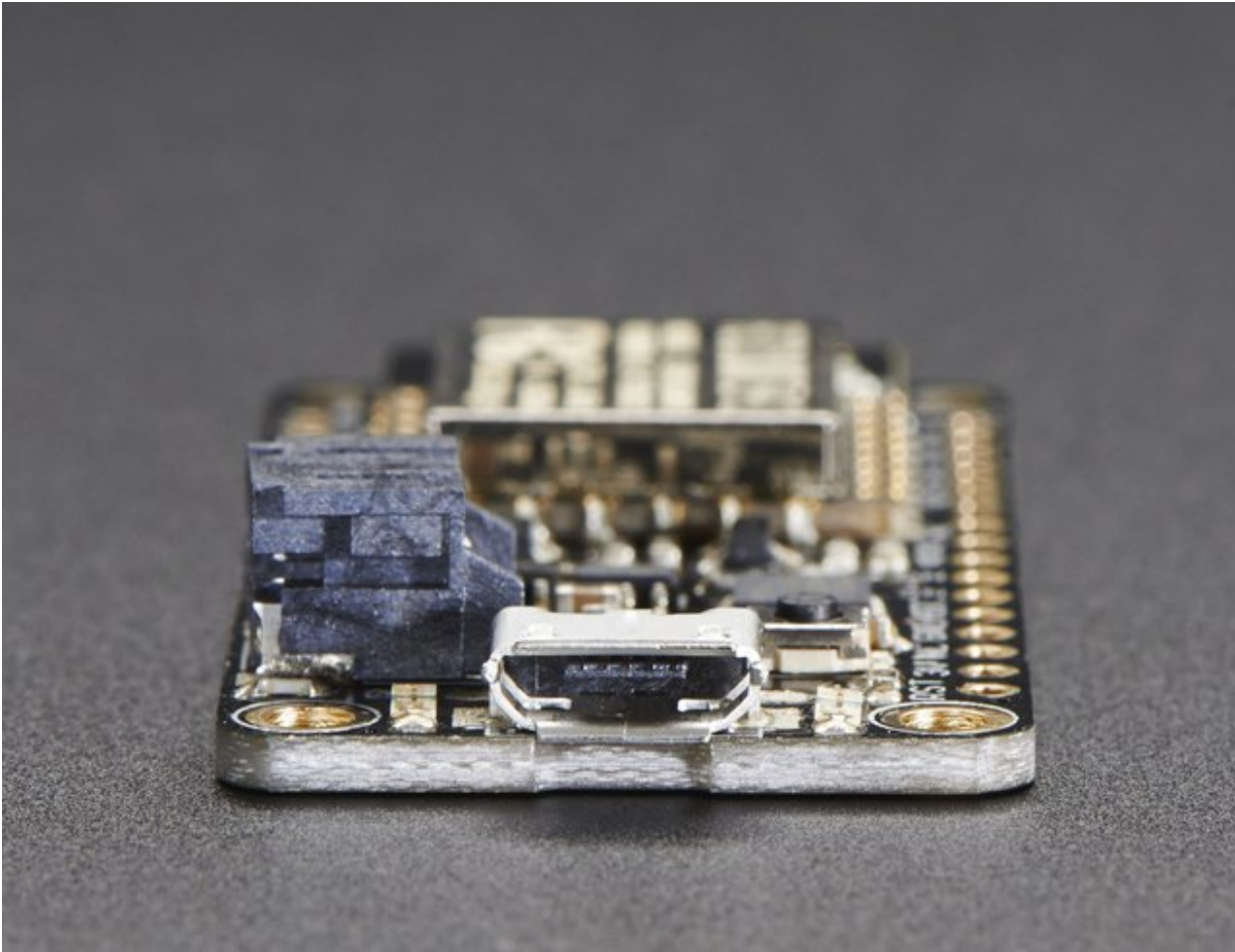


Feather is the new development board from Adafruit, and like it's namesake it is thin, light, and lets you fly! We designed Feather to be a new standard for portable microcontroller cores.

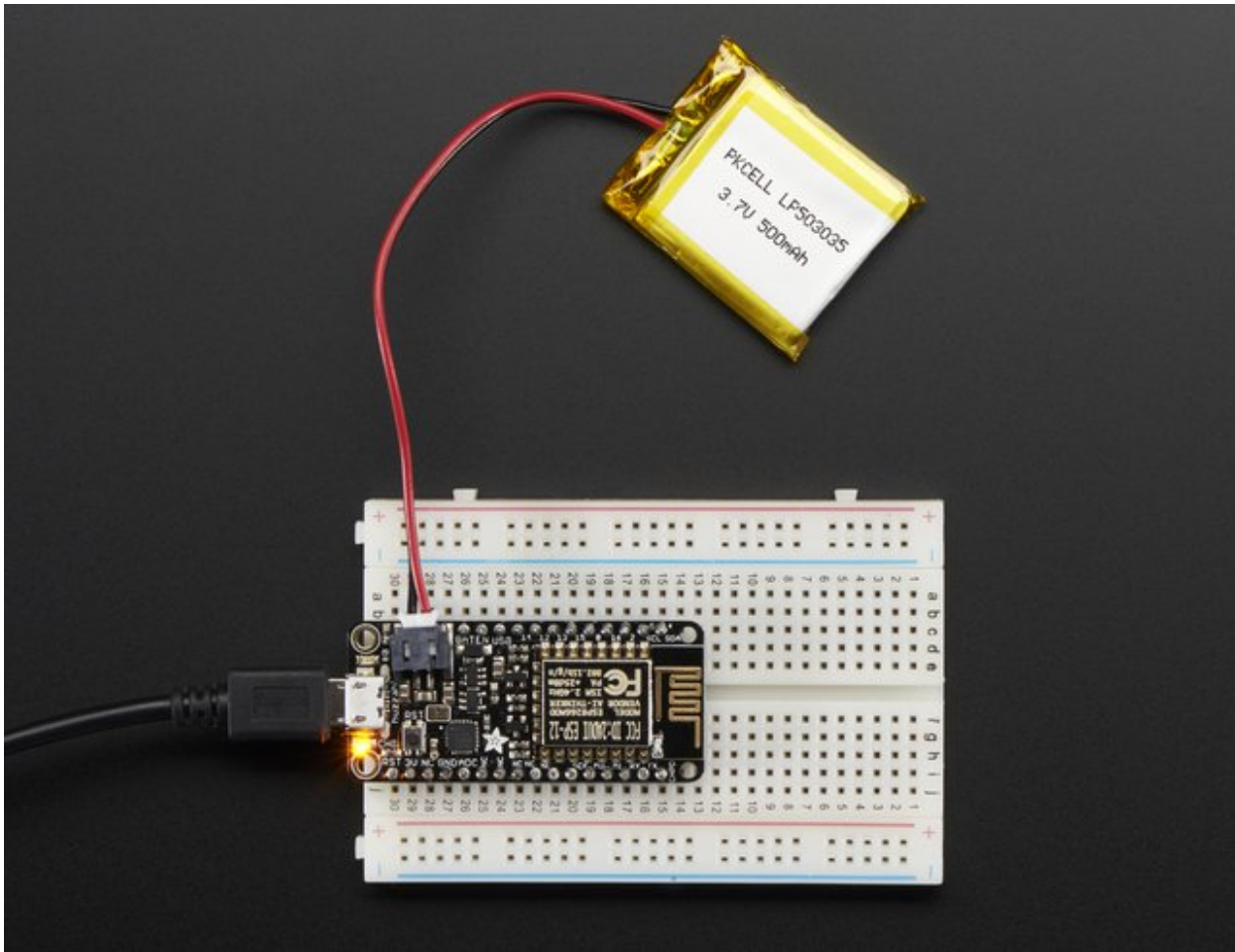
This is the **Adafruit Feather HUZAH ESP8266** - our take on an 'all-in-one' ESP8226 WiFi development board with built in USB and battery charging. Its an ESP8266 WiFi module with all the extras you need, ready to rock! [We have other boards in the Feather family, check'em out here \(http://adafru.it/jAQ\).](http://adafru.it/jAQ)



At the Feather HUZAZH's heart is an ESP8266 WiFi microcontroller clocked at 80 MHz and at 3.3V logic. This microcontroller contains a Tensilica chip core as well as a full WiFi stack. You can program the microcontroller using the Arduino IDE for an easy-to-run Internet of Things core. We wired up a USB-Serial chip that can upload code at a blistering 921600 baud for fast development time. It also has auto-reset so no noodling with pins and reset button pressings.

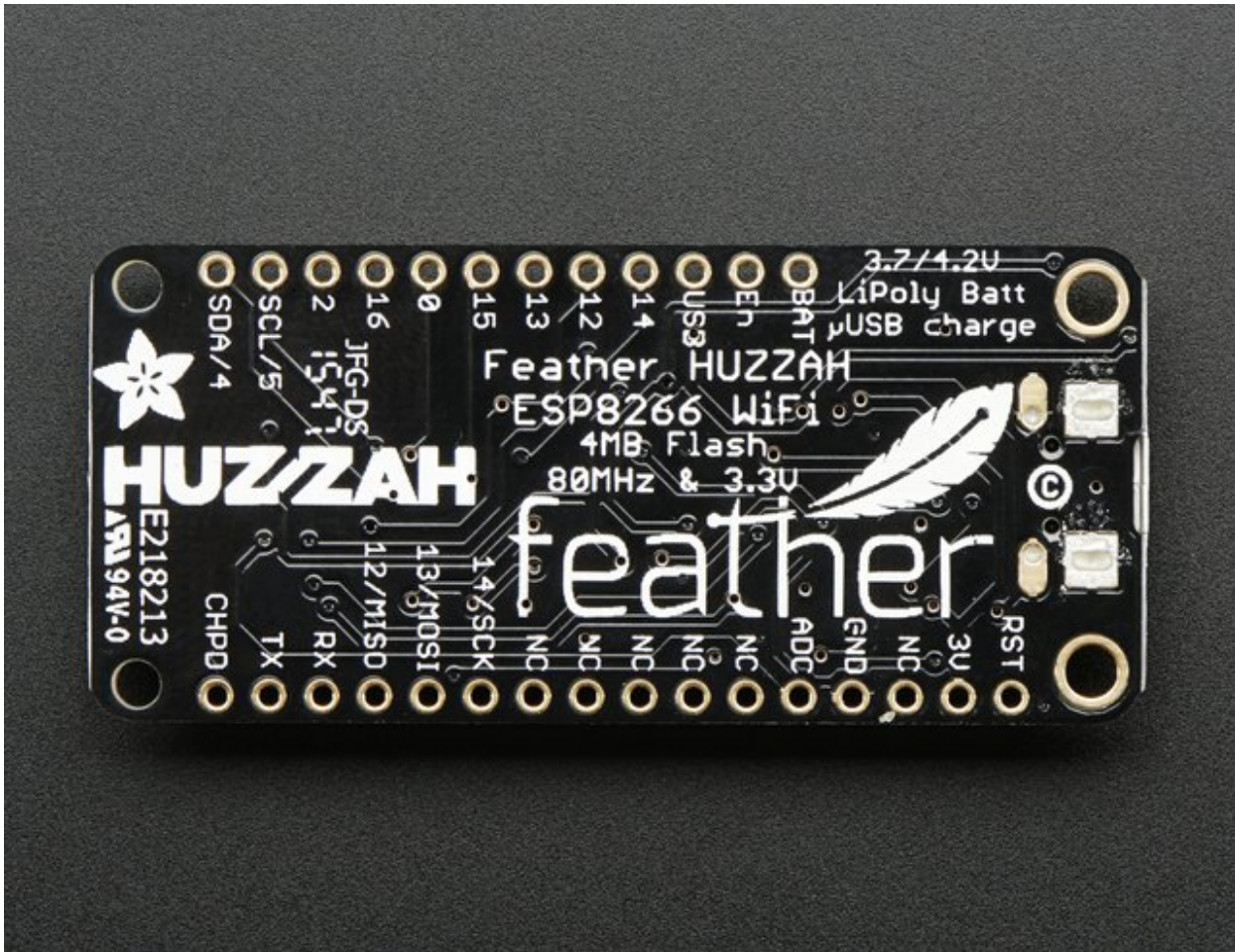


To make it easy to use for portable projects, we added a connector for any of our 3.7V Lithium polymer batteries and built in battery charging. You don't need a battery, it will run just fine straight from the micro USB connector. But, if you do have a battery, you can take it on the go, then plug in the USB to recharge. The Feather will automatically switch over to USB power when its available.



Here's some handy specs!

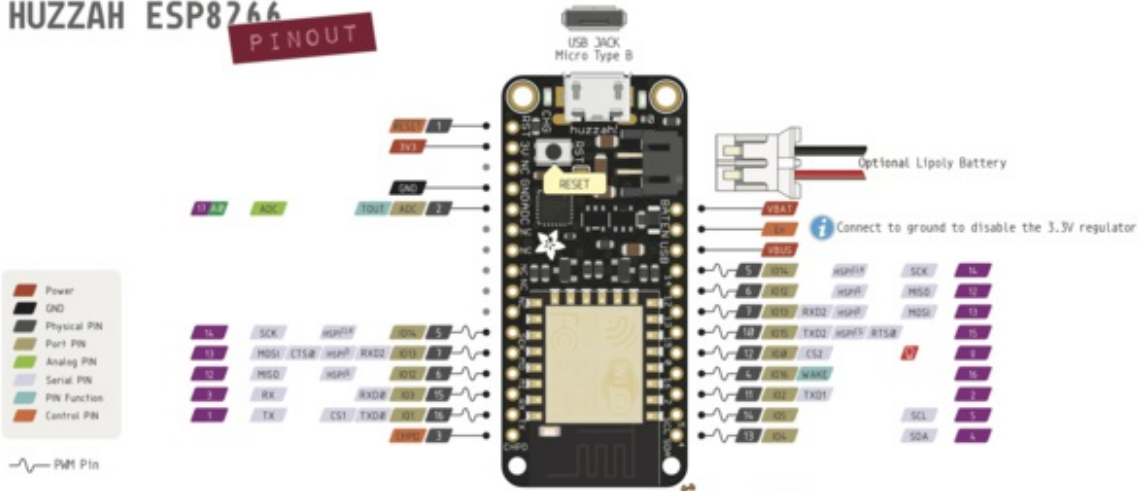
- Measures 2.0" x 0.9" x 0.28" (51mm x 23mm x 8mm) without headers soldered in
- Light as a (large?) feather - 6 grams
- ESP8266 @ 80MHz or 160 MHz with 3.3V logic/power
- 4MB of FLASH (32 MBit)
- 3.3V regulator with 500mA peak current output
- CP2104 USB-Serial converter onboard with 921600 max baudrate for uploading
- Auto-reset support for getting into bootload mode before firmware upload
- 9 GPIO pins - can also be used as I2C and SPI
- 1 x analog inputs 1.0V max
- Built in 100mA lipoly charger with charging status indicator LED
- Pin #0 red LED for general purpose blinking. Pin #2 blue LED for bootloading debug & general purpose blinking
- Power/enable pin
- 4 mounting holes
- Reset button



Comes fully assembled and tested, with a USB interface that lets you quickly use it with the Arduino IDE or NodeMCU Lua. (It comes preprogrammed with the Lua interpreter) We also toss in some header so you can solder it in and plug into a solderless breadboard. **Lipoly battery and USB cable not included** (but we do have lots of options in the shop if you'd like!)



Pinouts



Caution! Max per pin 12mA, 5mA recommended

Caution! Max per pin 85mA for the entire package

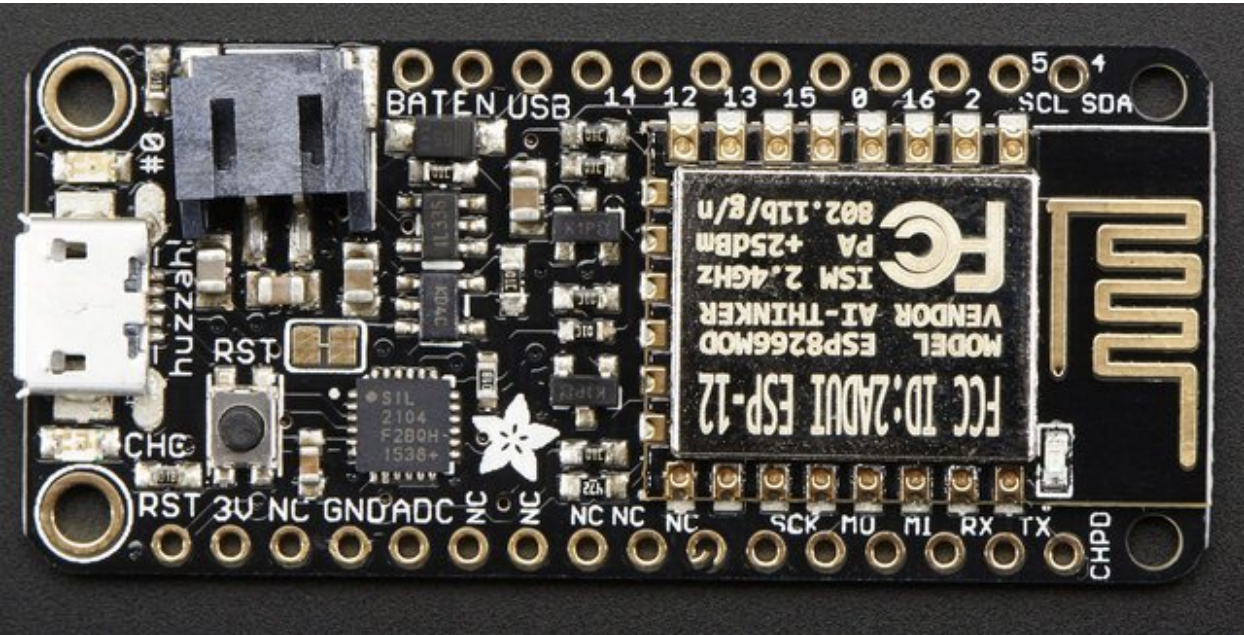
VBAT Connected to 5V USB Port
Max 500mA

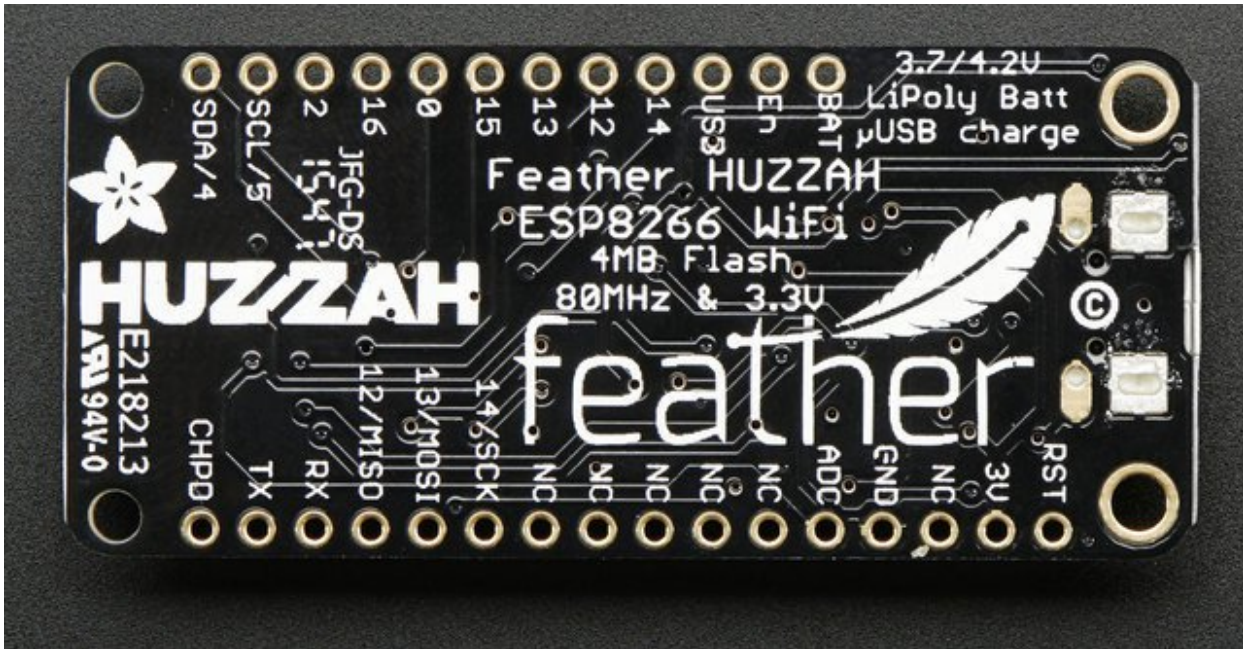
VBUS It's the positive voltage from to 3.3V Batt Jack

3V3 3V3 output from regulator
Max 400mA

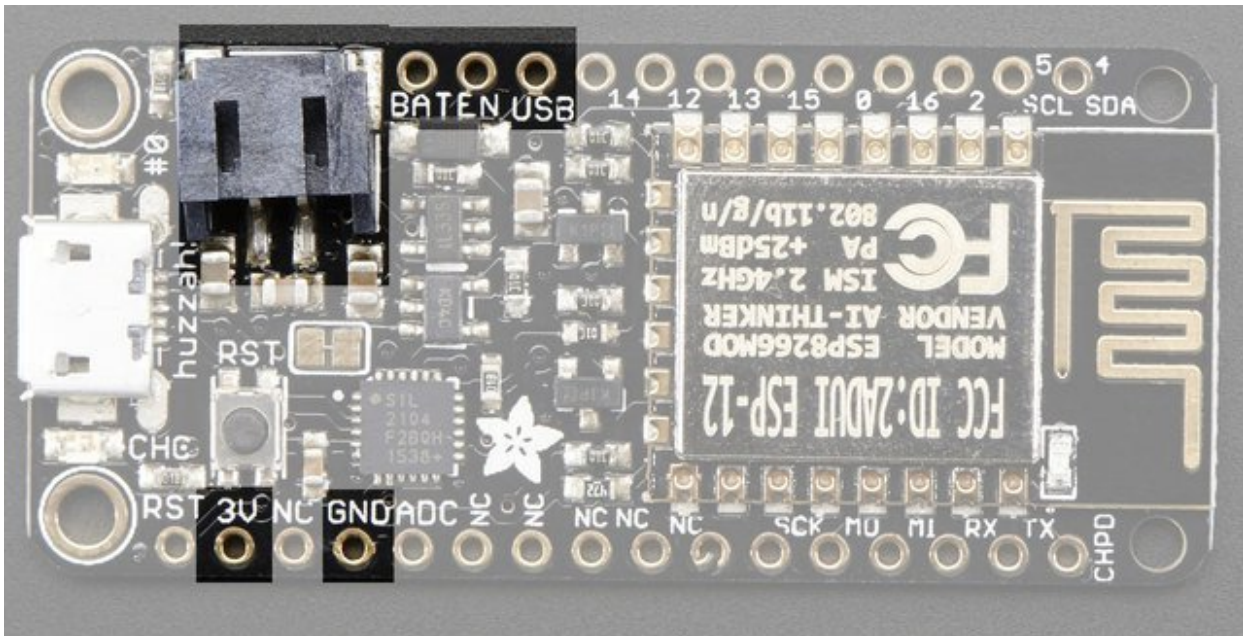


<https://www.adafruit.com/product/2821>





Power Pins



- **GND** - this is the common ground for all power and logic
- **BAT** - this is the positive voltage to/from the JST jack for the optional Lipoly battery
- **USB** - this is the positive voltage to/from the micro USB jack if connected
- **EN** - this is the 3.3V regulator's enable pin. It's pulled up, so connect to ground to disable the 3.3V regulator
- **3V** - this is the output from the 3.3V regulator, it can supply 500mA peak (try to keep your current draw under 250mA so you have plenty for the ESP8266's power requirements!)

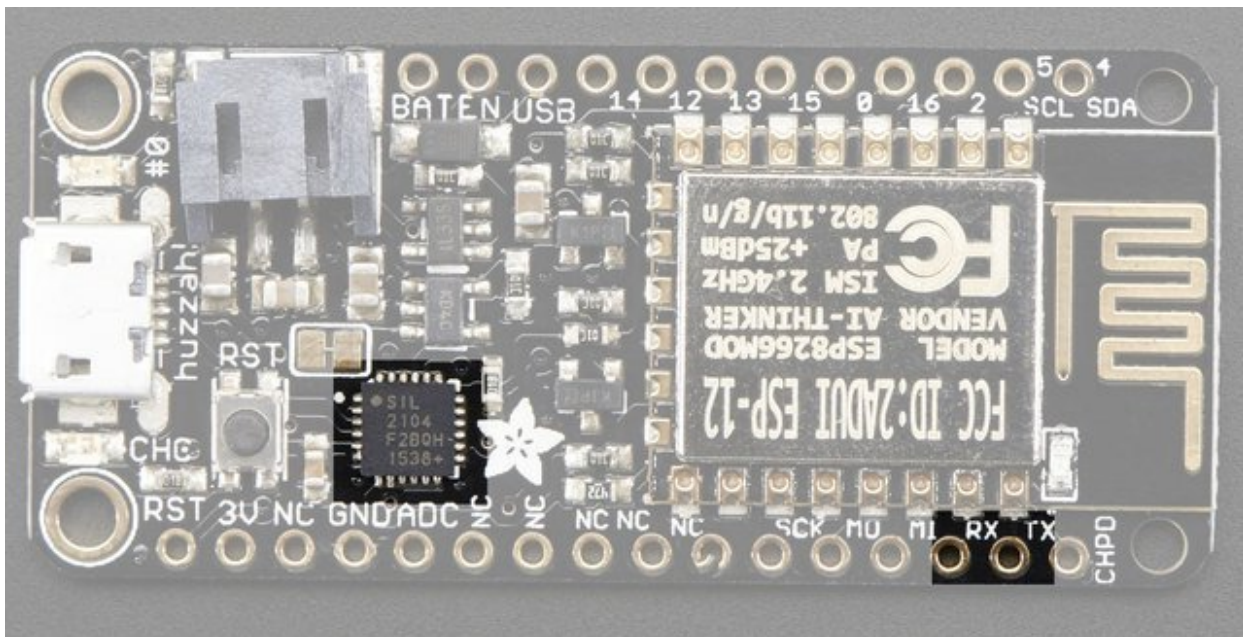
Logic pins

This is the general purpose I/O pin set for the microcontroller. All logic is 3.3V

The ESP8266 runs on 3.3V power and logic, and unless otherwise specified, GPIO pins are not 5V safe! The analog pin is also 1.0V max!

Serial pins

RX and **TX** are the serial control and bootloading pins, and are how you will spend most of your time communicating with the ESP module



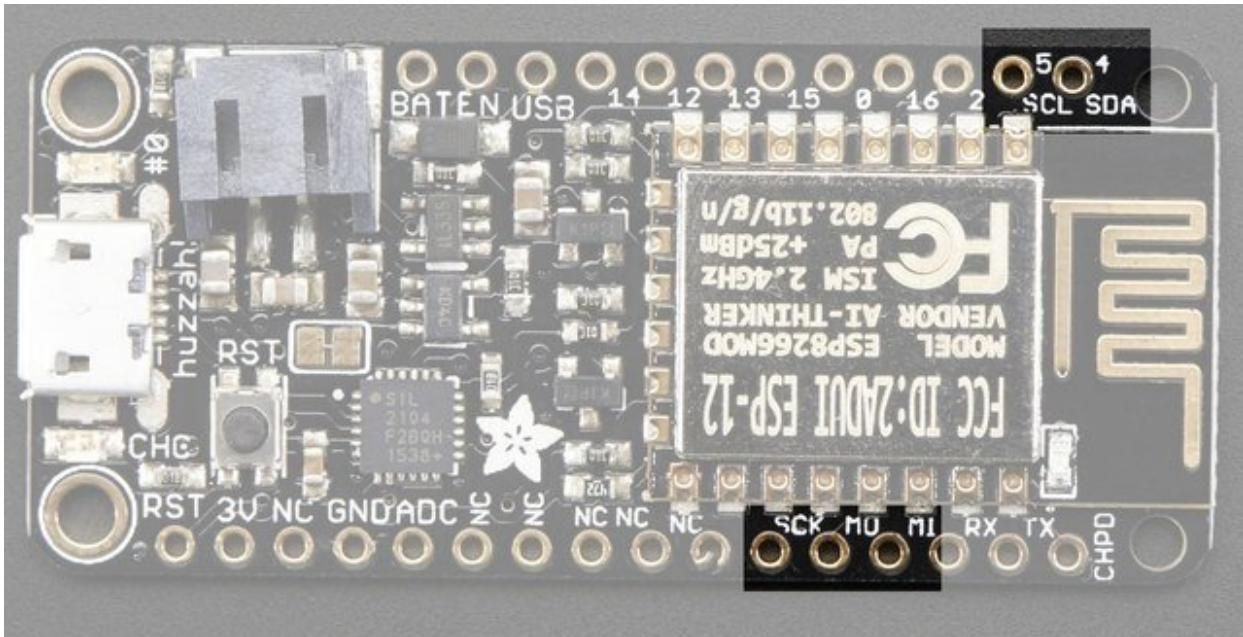
The **TX** pin is the output *from* the module and is 3.3V logic.

The **RX** pin is the input *into* the module and is 5V compliant (there is a level shifter on this pin)

These are connected through to the CP2104 USB-to-Serial converter so they should *not* be connected to or used unless you're super sure you want to because you will also be getting the USB traffic on these!

I2C & SPI pins

You can use the ESP8266 to control I2C and SPI devices, sensors, outputs, etc. While this is done by 'bitbanging', it works quite well and the ESP8266 is fast enough to match 'Arduino level' speeds.



In theory you can use *any* pins for I2C and SPI but to make it easier for people using existing Arduino code, libraries, sketches we set up the following:

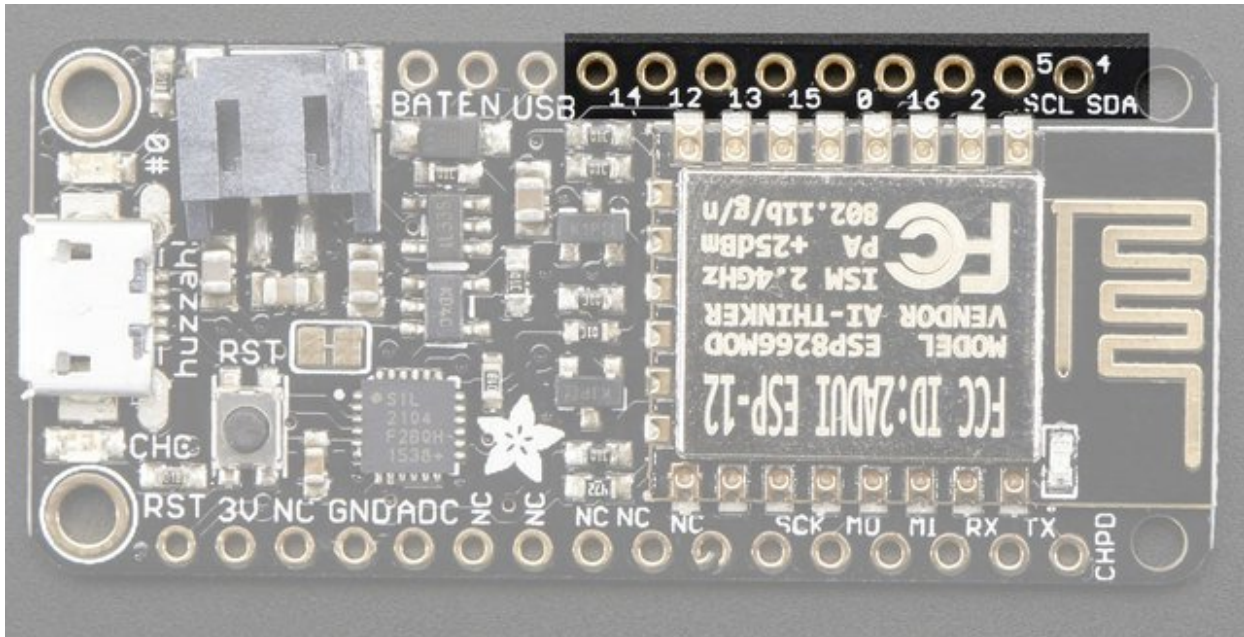
- **I2C SDA = GPIO #4** (default)
- **I2C SCL = GPIO #5** (default)

If you want, you can connect to I2C devices using other 2 pins in the Arduino IDE, by calling `Wire.pins(sda, scl)` before any other Wire code is called (so, do this at the beginning of `setup()` for example

Likewise, you can use SPI on any pins but if you end up using 'hardware SPI' you will want to use the following:

- **SPI SCK = GPIO #14** (default)
- **SPI MOSI = GPIO #13** (default)
- **SPI MISO = GPIO #12** (default)

GPIO pins



This breakout has 9 GPIO: **#0, #2, #4, #5, #12, #13, #14, #15, #16** arranged at the top edge of the Feather PCB

All GPIO are 3.3V logic level in and out, and are **not 5V compatible**. Read the [full spec sheet](http://adafruit.it/f1E) (<http://adafruit.it/f1E>) to learn more about the GPIO pin limits, but be aware the maximum current drawn per pin is **12mA**.

These pins are general purpose and can be used for any sort of input or output. Most also have the ability to turn on an internal pullup. Many have *special* functionality:

GPIO #0, which does not have an internal pullup, and is also connected a red LED. This pin is used by the ESP8266 to determine when to boot into the bootloader. If the pin is held low during power-up it will start bootloading! That said, you can always use it as an output, and blink the red LED.

GPIO #2, is also used to detect boot-mode. It also is connected to the blue LED that is near the WiFi antenna. It has a pullup resistor connected to it, and you can use it as any output (like #0) and blink the blue LED.

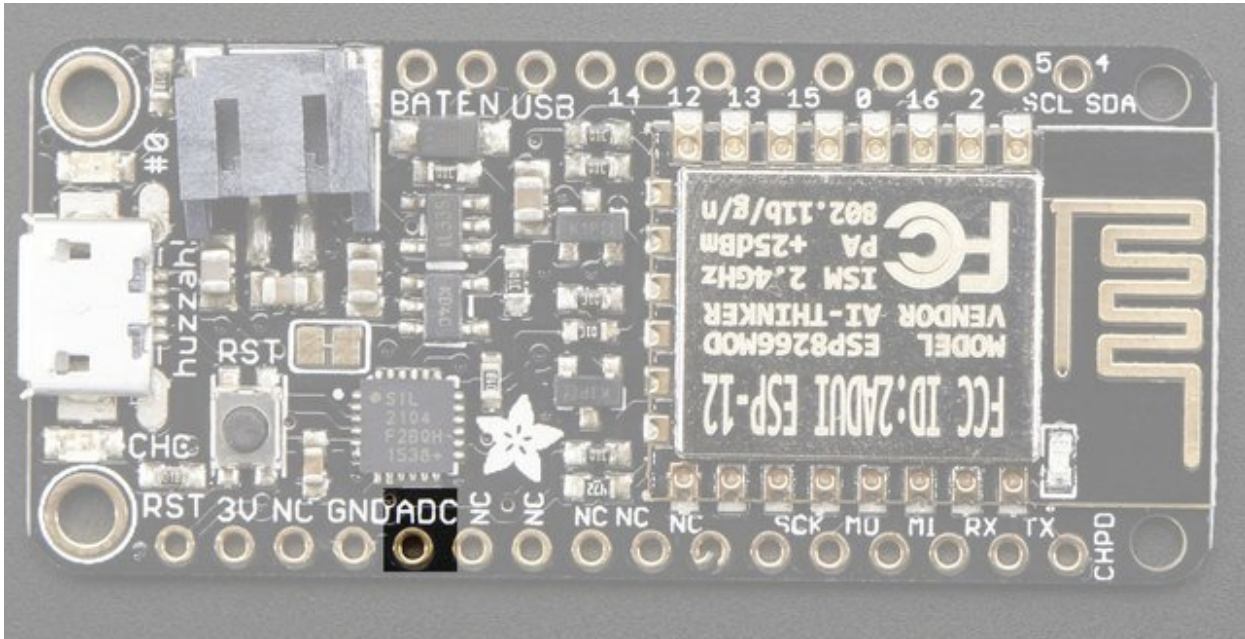
GPIO #15, is also used to detect boot-mode. It has a pulldown resistor connected to it, make sure this pin isn't pulled high on startup. You can always just use it as an output

GPIO #16 can be used to wake up out of deep-sleep mode, you'll need to connect it to the RESET pin

Also note that GPIO **#12/13/14** are the same as the **SCK/MOSI/MISO** 'SPI' pins!

Analog Pins

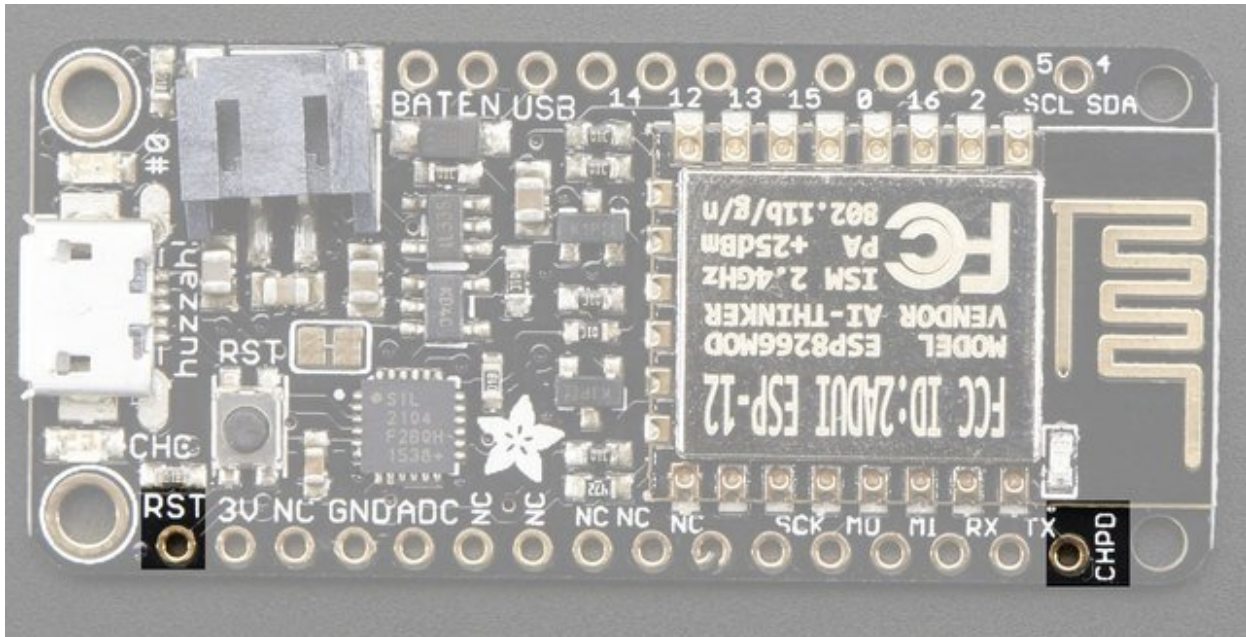
There is also a single analog input pin called **A**. This pin has a ~1.0V maximum voltage, so if you have an analog voltage you want to read that is higher, it will have to be divided down to 0 - 1.0V range



Other control pins

We have a few other pins for controlling the ESP8266

- **RST** - this is the reset pin for the ESP8266, pulled high by default. When pulled down to ground momentarily it will reset the ESP8266 system. This pin is 3.3V logic only
- **EN (CH_PD)** - This is the enable pin for the ESP8266, pulled high by default. When pulled down to ground momentarily it will reset the ESP8266 system. This pin is 3.3V logic only



NC Pins

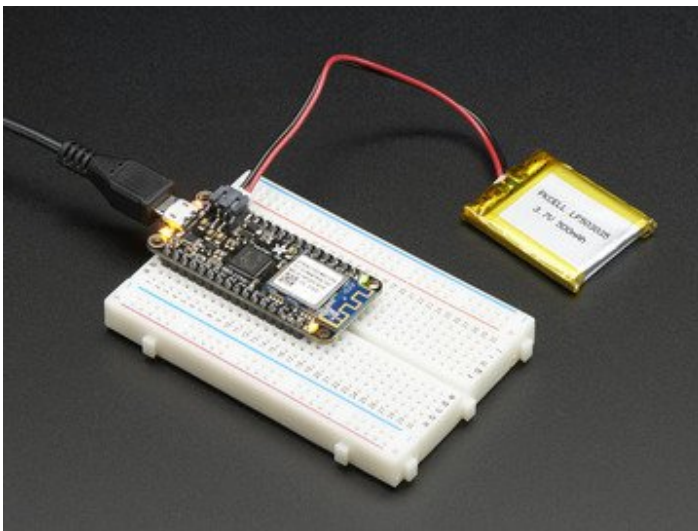
The rest of the pins are labeled **NC** which means **Not Connected** - they are not connected to anything and are there as placeholders only, to maintain physical compatibility with the other boards in the Feather line!

Assembly

We ship Feathers fully tested but without headers attached - this gives you the most flexibility on choosing how to use and configure your Feather

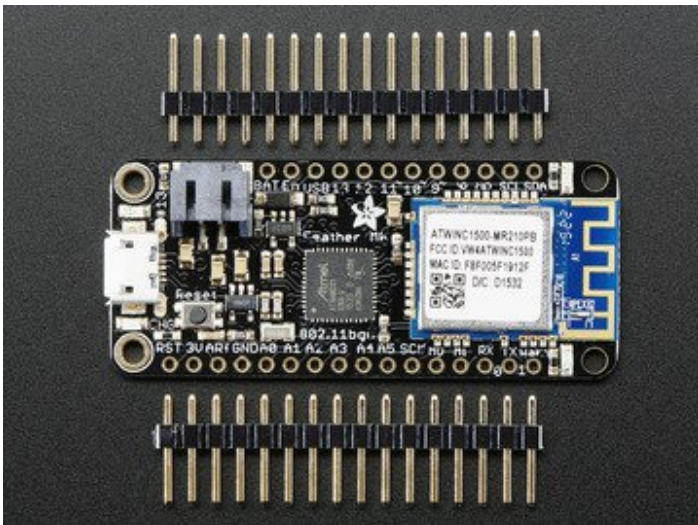
Header Options!

Before you go gung-ho on soldering, there's a few options to consider!

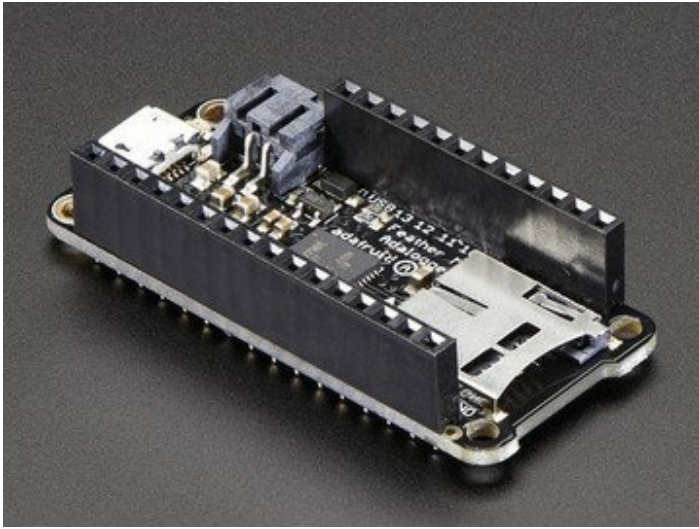


-

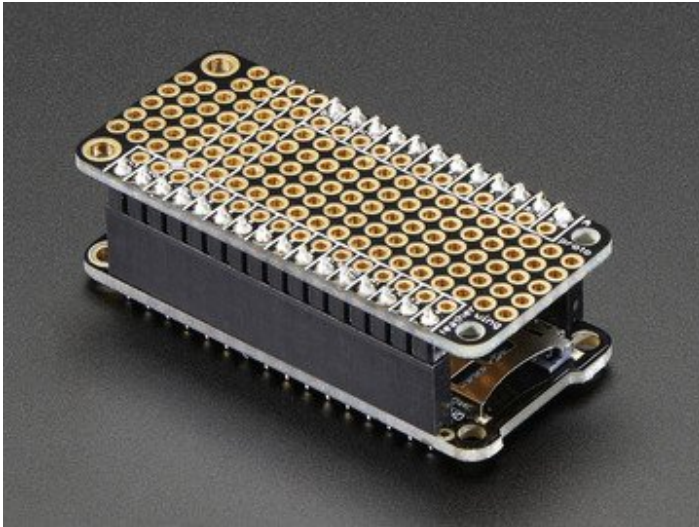
The first option is soldering in plain male headers, this lets you plug in the Feather into a solderless breadboard

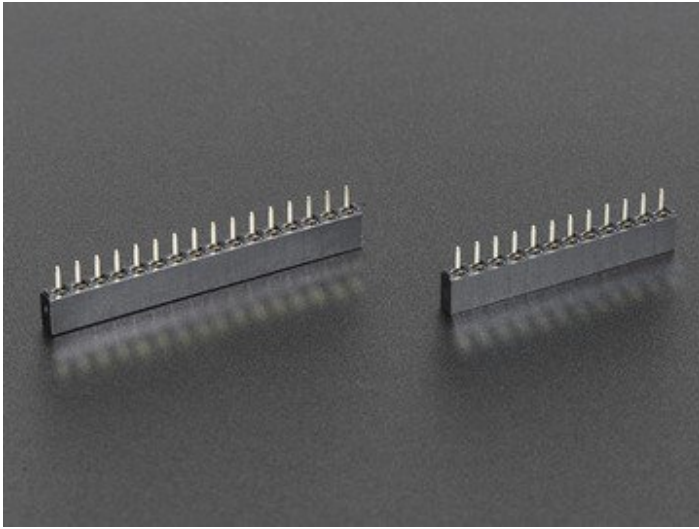


-

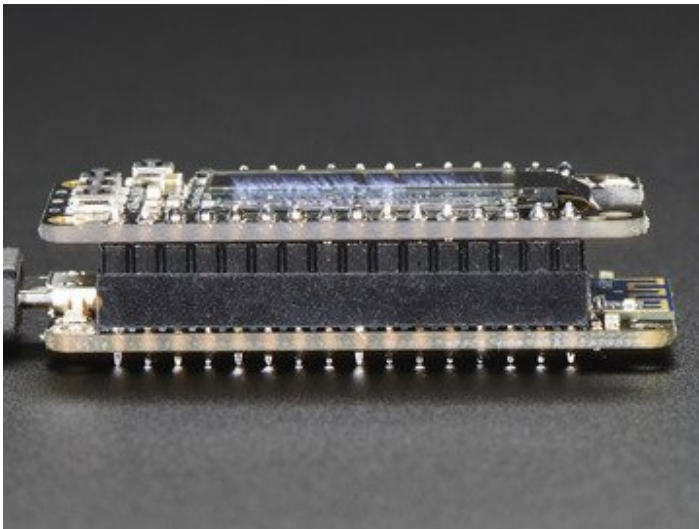


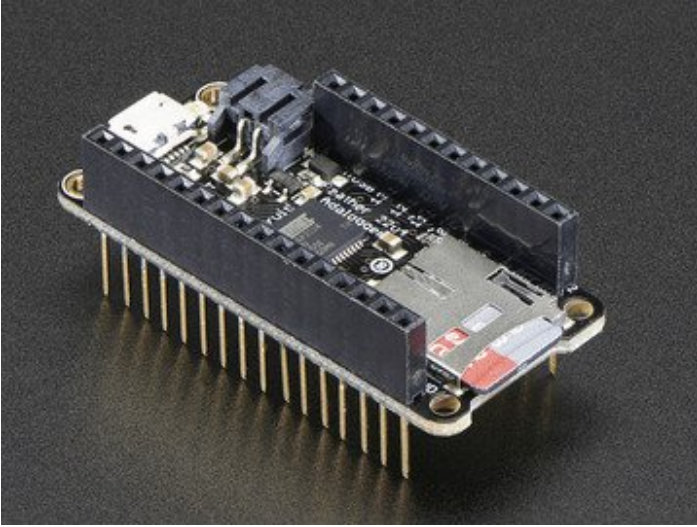
Another option is to go with socket female headers. This won't let you plug the Feather into a breadboard but it will let you attach featherwings very easily



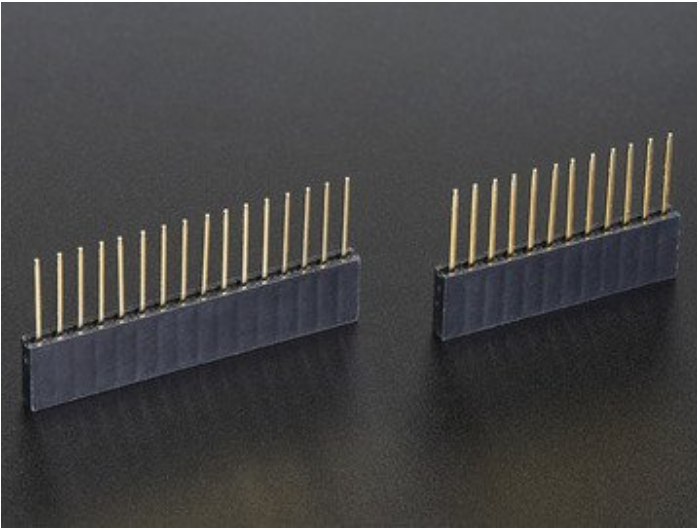


We also have 'slim' versions of the female headers, that are a little shorter and give a more compact shape

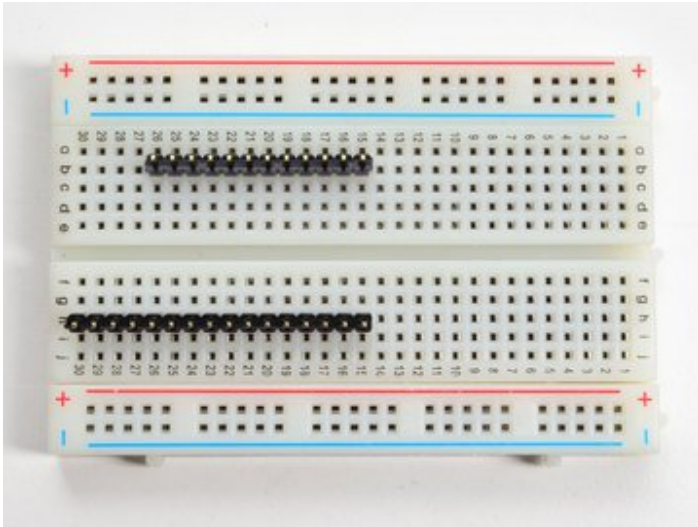




Finally, there's the "Stacking Header" option. This one is sort of the best-of-both-worlds. You get the ability to plug into a solderless breadboard *and* plug a featherwing on top. But its a little bulky

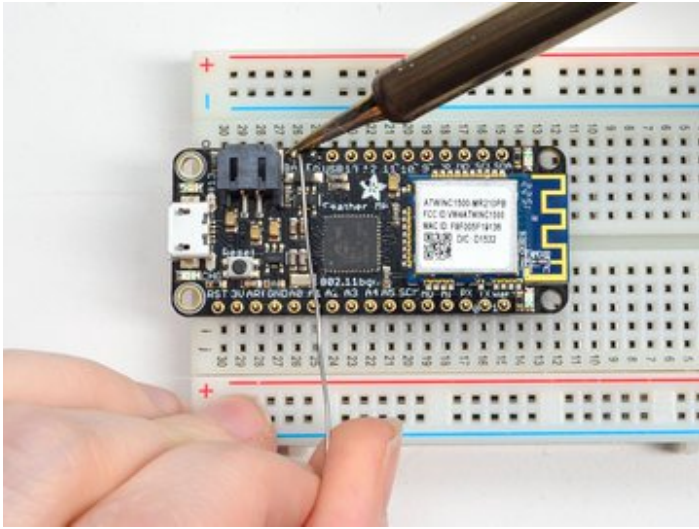


Soldering in Plain Headers



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**

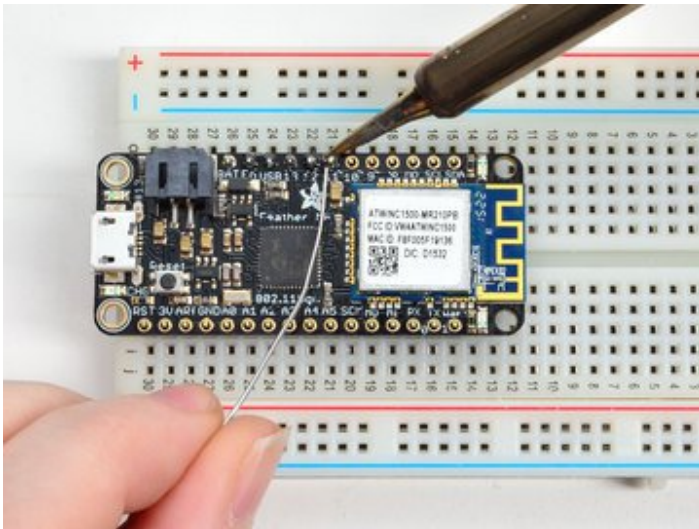


Add the breakout board:

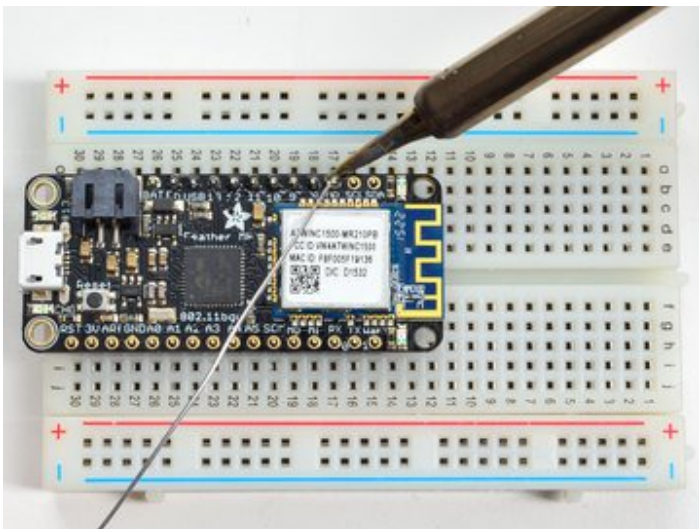
Place the breakout board over the pins so that the short pins poke through the breakout pads

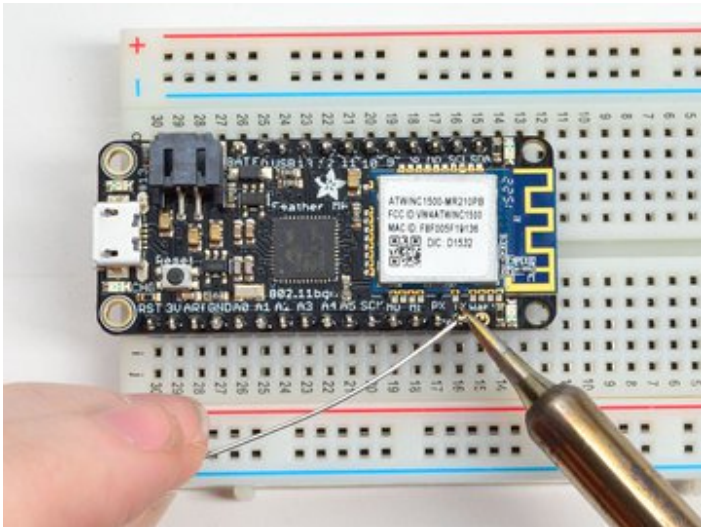
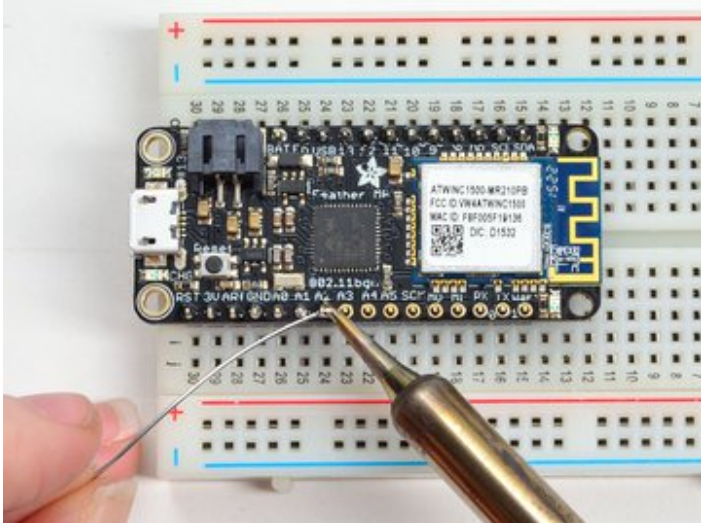
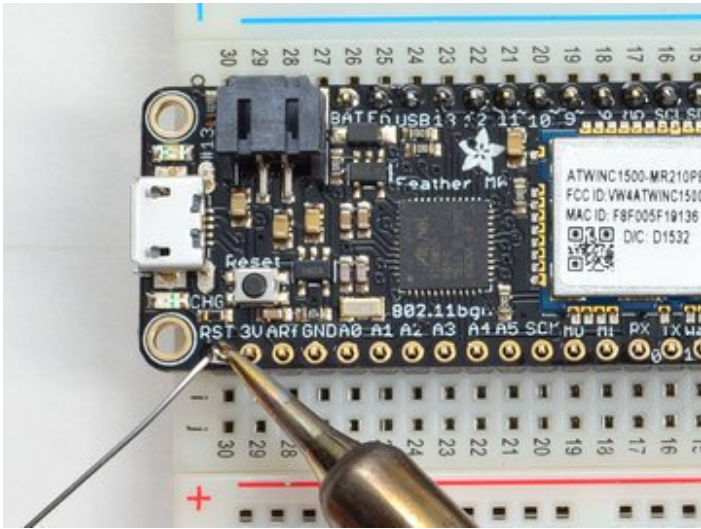
And Solder!

Be sure to solder all pins for reliable electrical contact.

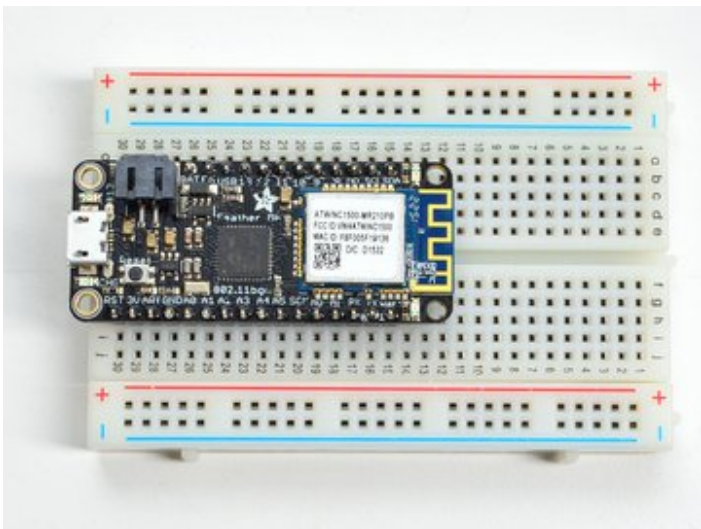


(For tips on soldering, be sure to check out our [Guide to Excellent Soldering](http://adafru.it/aTk) (<http://adafru.it/aTk>)).





Solder the other strip as well.



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

Soldering on Female Header



Tape In Place

For sockets you'll want to tape them in place so when you flip over the board they don't fall out

Flip & Tack Solder

After flipping over, solder one or two points on each strip, to 'tack' the header in place

