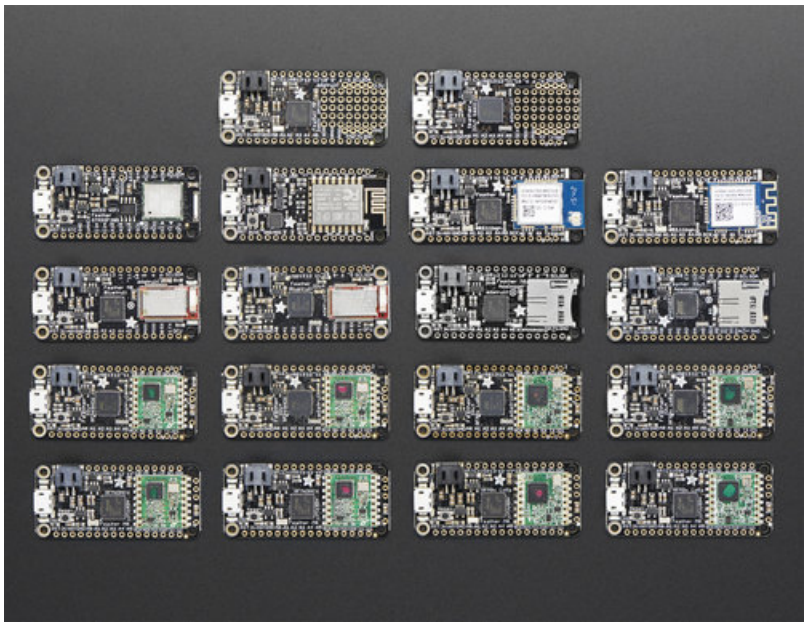
adafruit learning system

Introducing Adafruit Feather

Created by lady ada



Last updated on 2019-02-11 02:55:22 AM UTC

Guide Contents

Overview

Wouldn't it be great if, instead of collecting shields, HATs, PMODs, Clicks, Booster Packs, Props... you could have a cross-compatible platform? Something that doesn't force you into one chipset? and allows you to mix-and-match the microcontroller, wireless protocol, and functional extensions?

Yeah, me too! That's why I created Feather. Feather is a flexible and powerful family of microcontroller main-boards (Feathers) and daughter-boards (Wings) with a wide range of capabilities.

Feathers!

Since we aren't tied to one chip manufacturer, we feature the best-of-the-market chipsets, including:

- Atmel ATmega32u4 and ATmega 32P - 8 bit AVR
- Atmel ATSAMD21 - 32 bit ARM Cortex M0+
- Atmel ATSAMD51 - 32-bit ARM Cortex M4
- Broadcom/Cypress WICED - STM32 with WiFi
- Espressif ESP8266 and ESP32 - Tensilica with WiFi/BT
- Freescale MK20 - ARM Cortex M4, as the Teensy 3.2 Feather Adapter
- Nordic nRF52832 and nRF32840 - ARM Cortex & Bluetooth LE
- Packet radio modules featuring SemTech SX1231
- LoRa radio modules featuring SemTech SX127x

and many more to come!

Wings!

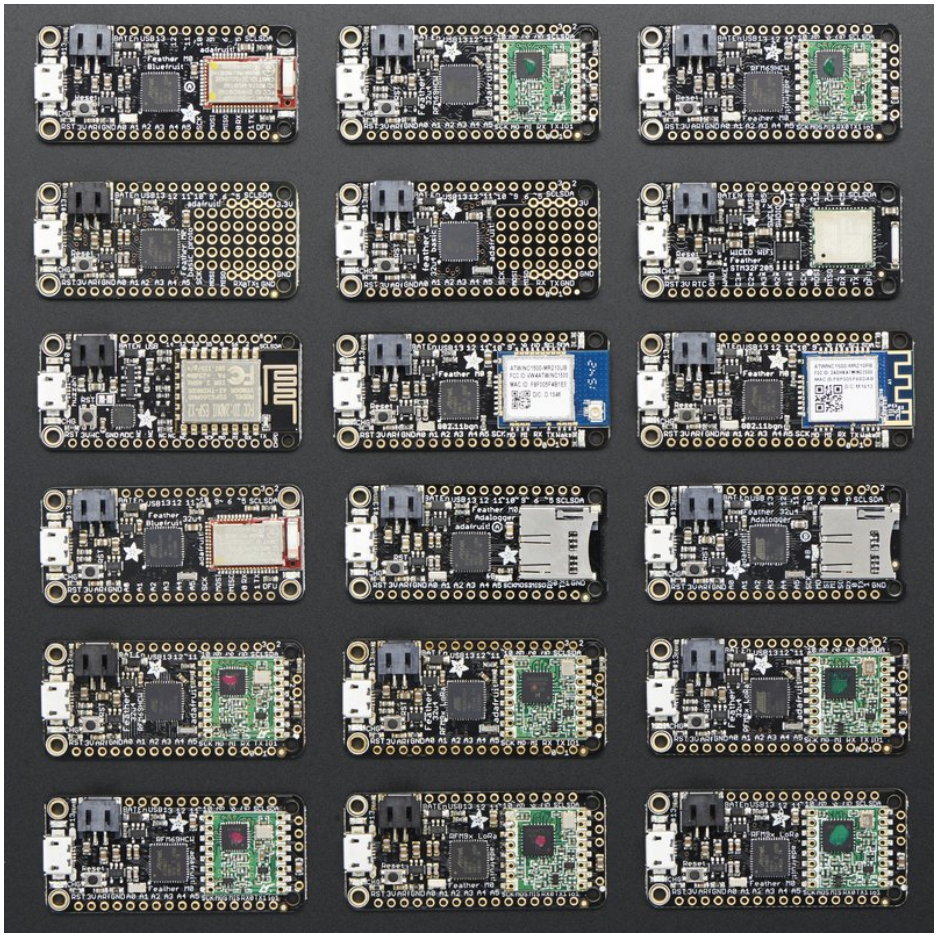
Once you've picked your main dish (the Feather mainboard) you can pick and choose from a wide wide variety of Wings. Wings are like little "shields" or "HATs" for Feathers. You can (in theory) add as many Wings as you like to a single Feather main-board, as long as you don't run out of power, space or have pin collisions.

Best of all, we've tested all of our Wings against all of our Feathers. So, other than a few exceptions (mostly the GPS Featherwing), you can re-use, re-cycle, and upgrade your Feathers and Wings between projects and as need arises.

For example, start out with a Feather 32u4, then decide to upgrade to the Feather M0 or M4 for more Flash - the pin out is the same! Or, make an IoT project with the ESP8266 and then super-power it with an ESP32 just by swapping out the main Feather. Maybe build a project with a Feather M0 Bluefruit, then realize you need more range so you switch to a Feather M0 LoRa.

The Feathers and Wings all have example code written in Arduino C/C++ so as your projects adapt, all your wings and code will come along with you. Boards that have names with the Express suffix also support CircuitPython for fast and

easy programming.



Feather History

For the last decade or so I've been making projects with Arduinos and compatibles. And, an Arduino is a great way to start a project, but after many years I started seeing the same 'issues' crop up:

- Arduinos are *big* - sure they're smaller than a computer but they're still chunky enough to make them not-so-wearable or hand-held
- To do stuff, you can add shields, but stacking shields get really tall to get any significant technology stacked on
- There's no wireless technology built in: now that WiFi, BTLE and cellular is ubiquitous, just about any project can take advantage of a wireless connection!
- Making an Arduino portable is hard - you need a 7-12VDC power supply, not a common rechargeable Lithium Polymer battery, and no built-in recharging

Some possibilities

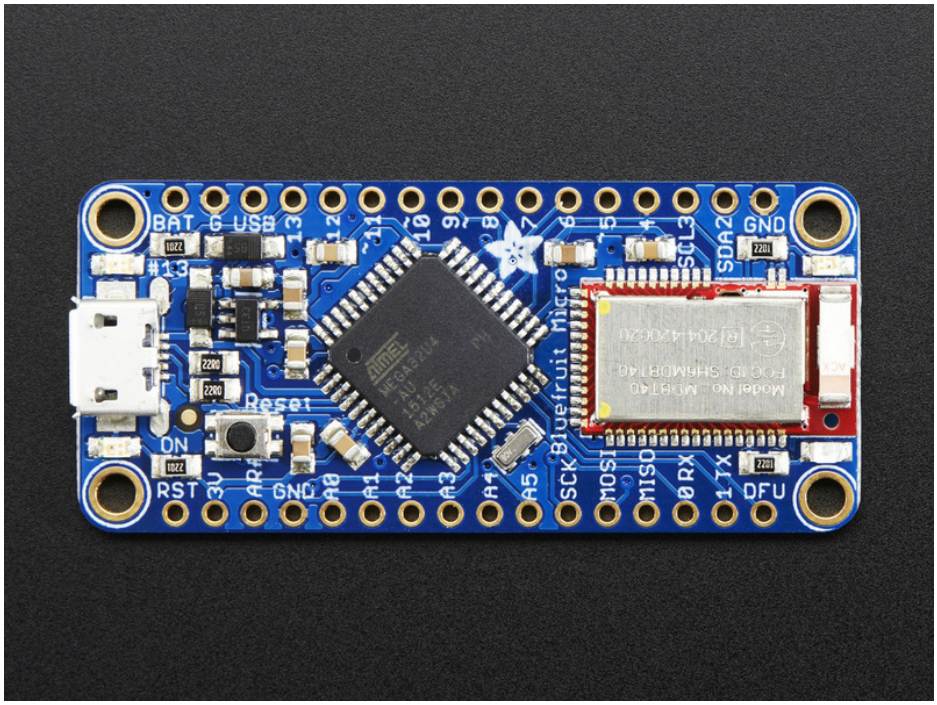
There are a few boards that tried to address these issues, but none of them really seemed to 'scratch that itch'

- Multi-stack compatibles like Tinyduino (https://adafru.it/wel) are small, and extensible, but you have to stack 4 boards just to get the 'basics' going
- Small Arduino-compatibles like the Pro Mini (https://adafru.it/weJ) are inexpensive and small, but don't have USB built in so you need a separate cable to program them
- The Fio (https://adafru.it/weK) has wireless add-on ability, and battery charging, but a non-standard pinout and requires expensive XBee modules.
- Click boards are close (https://adafru.it/weL) but don't have an Arduino compatible baseboard.

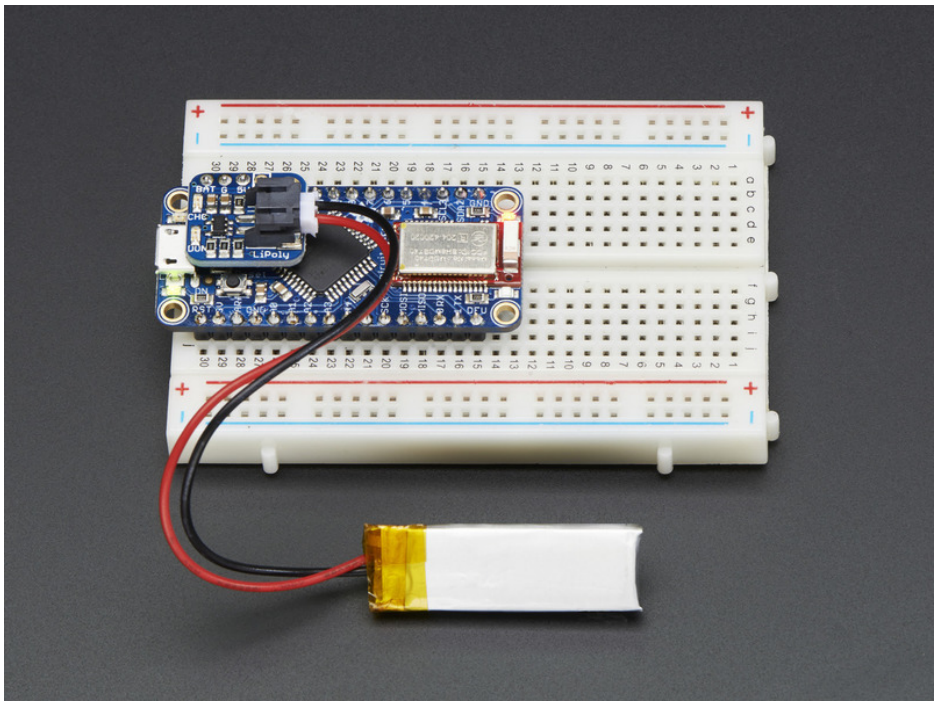
Beginning of a Feather

It was about when KTOWN had finished designing the SPI Bluefruit Friend when we kept (https://adafru.it/weM) seeing (https://adafru.it/weN) crowdfunding (https://adafru.it/weO)campaigns that had an Arduino-compatible chip and a Bluetooth LE interface. We thought "it would be really cool and useful to have something that was breadboard friendly, with an Atmega32u4 so it has built in USB, and a Bluefruit module. We've already got the library code, it'll be an all-in-one!"

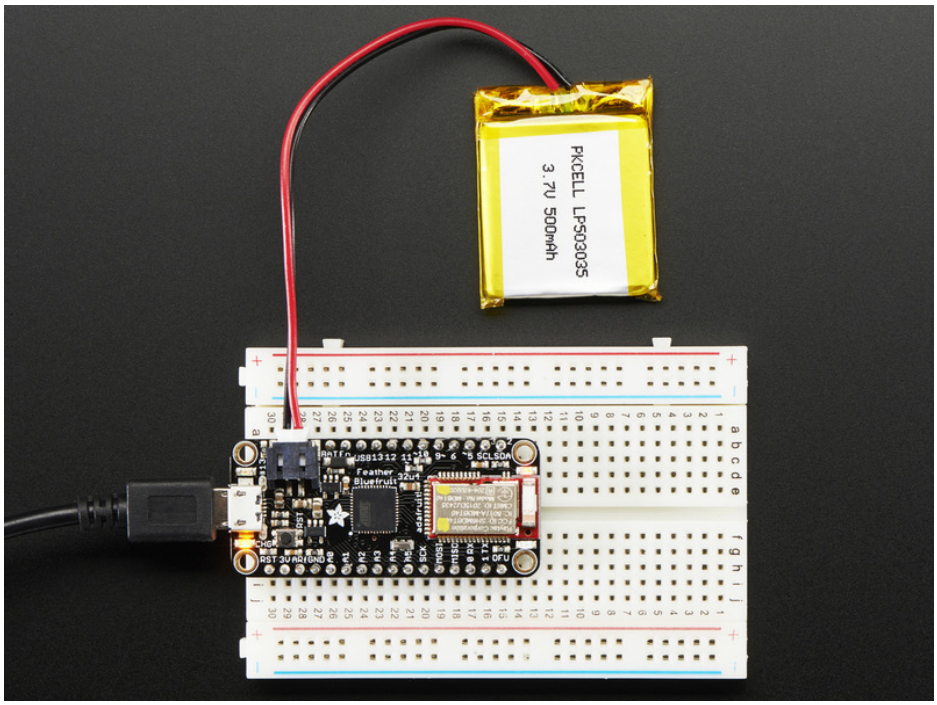
So we designed the Bluefruit Micro (https://adafru.it/fQH)



We designed it so you could use our little LiPoly backpack, soldered on top, to add LiPo power and recharging:



The Bluefruit Micro was incredibly popular. So much so that I realized we should probably do something better. So I shrank the chip from a QFP to a QFN and stuck the battery connector off the side:



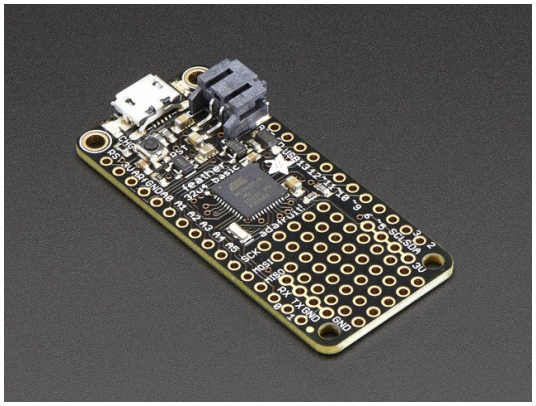
As you can see, not much in the design really changed, just some parts got squished down and pushed to the right to make room for the LiPoly charge and connector.

I settled on a pinout configuration that exposed all the power pins, analog inputs, GPIO, I2C, SPI and UART. Once that was done I tried making a few different 'flavors' such as with a micro SD card holder to make a datalogger:

Your browser does not support the video tag. Adafruit Feather 32u4 Adalogger

$21.95
IN STOCK
ADD TO CART

Or just a prototyping area:

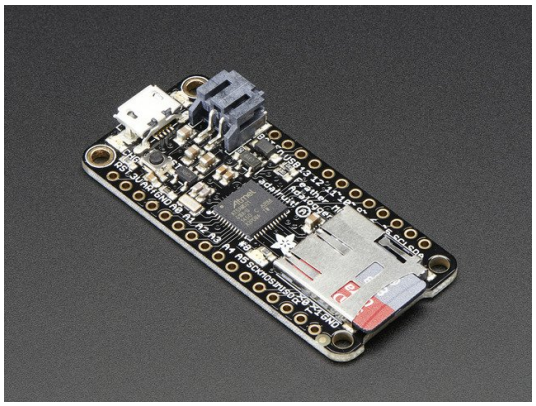


Adafruit Feather 32u4 Basic Proto

$19.95
IN STOCK
ADD TO CART

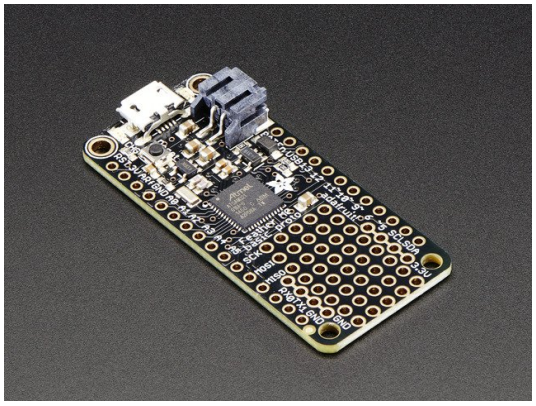
An M0 Challenger Appears!

A new chip in the market at the time was the ATSAMD21, used in the Arduino Zero. This chip is about the same price as the ATmega32u4 but has 8x as much FLASH, 8x as much RAM, is a full 32 bit ARM Cortex M0+, runs 6 times faster and has a bazillion more peripherals including a DAC! Feather turned out to be a great way to make a board using this nice new chip. So I designed versions of Adalogger, Basic and Bluefruit to match!



Adafruit Feather M0 Adalogger

$19.95
IN STOCK
ADD TO CART



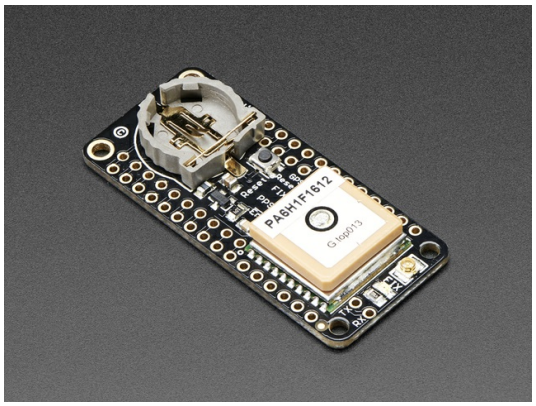
Adafruit Feather M0 Basic Proto - ATSAMD21 Cortex M0

$19.95
IN STOCK
ADD TO CART

Wing Friendz

Things were looking pretty good, I had a favorable view of the size, pinouts, and capabilities. So I started thinking about 'add ons' - little shields that could add capability. For example, I knew I'd need something to add GPS capability. So I designed a board that connected to the UART pins and would fit in the same shape as the feather. That managed

to fit, and even had room for a battery backup:



Adafruit Ultimate GPS FeatherWing

$39.95
IN STOCK

ADD TO CART

Then, I took the most popular Arduino shield we have, the motor shield, and squeezed it down to a Wing. That fit fine too!

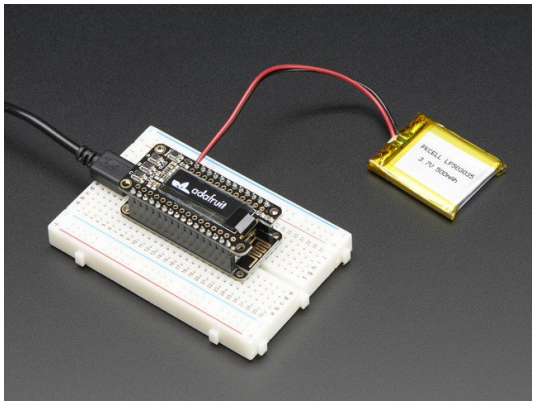
Your browser does not support the video tag.

DC Motor + Stepper FeatherWing Add-on For All Feather Boards

$19.95
IN STOCK

ADD TO CART

I tried designing a couple more FeatherWings, using I2C, SPI, GPIO etc. Like this OLED wing:



Adafruit FeatherWing OLED - 128x32 OLED Add-on For Feather

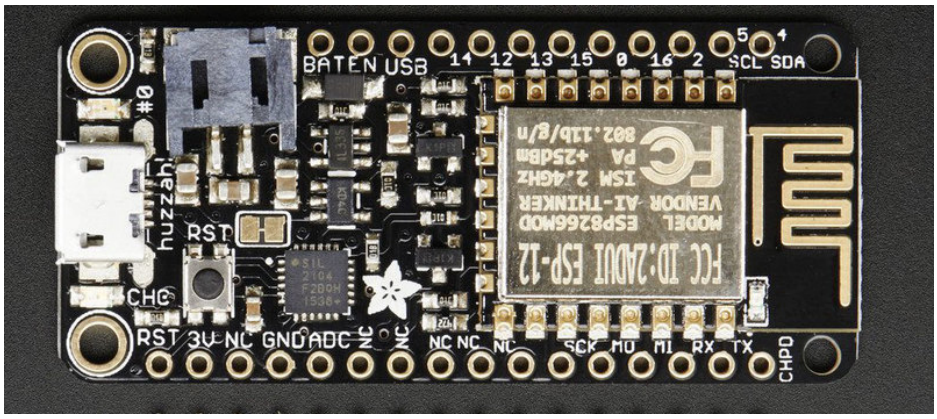
$14.95
IN STOCK

ADD TO CART

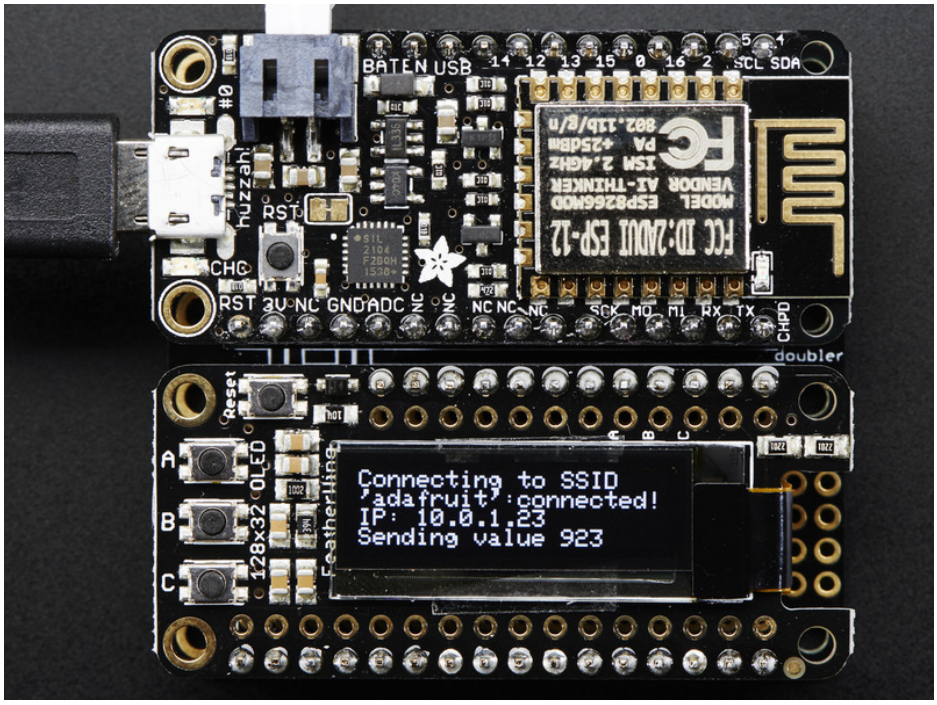
By this time I had maybe a dozen or so Feathers and Wings. So far I'd only been using the ATmega32u4 and ATSAMD21. These chipsets are powerful and have plenty of capability. But to make the ecosystem really work I needed to make the most 'difficult' and 'minimal' Feather, to verify all my Wing designs would still work

ESP8266 Feather

Once I had a prototype of this design I sort of realized that the basic idea could be extended to other modules, not just the nRF51 Bluefruit module. For example, at the time, the ESP8266 was gaining in popularity, and there was a module that had the chip, flash memory and antenna. We already had a breakout board but it didn't have a USB interface, or battery charger. Adding a USB to serial converter and a lipoly charger would make the board a little bigger, and with some arrangement, I fit everything in the same 0.9" by 2.0"



Since the ESP8266 was so constrained: only one 1.0V ADC, few pins, odd pullups and pulldowns, it was a perfect test bed for all the Wings I could create. I started cross-testing all the Wings with this chip. If I could get them working with this Feather and the 32u4 and the SAMD21 that covered 3 different architectures (Tensilica, 8-bit AVR, 32-bit ARM) for good functional coverage.



From then on, it was easy going. I just designed a Feather or Wing every single week for a full year. I always knew what I'd work on next because it was just a Feather or accessory. Each new Wing or Feather gets tested against all the other accessories to make sure they all work. It's a little tedious but that's the great thing about Feather, is the full cross-compatibility. Now, every time a new chip comes out, I don't have to "start all over from scratch". For example, designing the ESP32 Feather was a little constraining to make it all fit but once it fit, I knew that I didn't have to make

an OLED add-on, terminal block add-on, LED matrix add-on, etc.

As of this writing, May 2017, there are 52 Wings and 26 main boards. Other companies and customers are also designing their own compatibles.

I fully expect to have 100 Wings and 50 main boards in the next year or two - Feather has been a great success!

Feather Specification

Are you interested in making something that works with the Adafruit Feather & FeatherWing system?

That's awesome! Adafruit encourages you to do so! Here's some guidelines to make it easy to make sure you can mix & match in the nearly-100-board ecosystem.

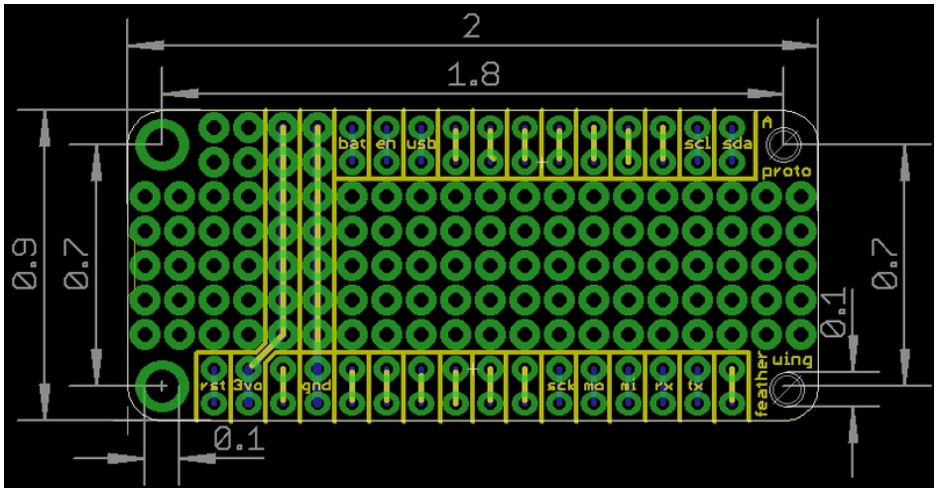
We will do our best to update this whenever we can!

Feather & Wing Sizes

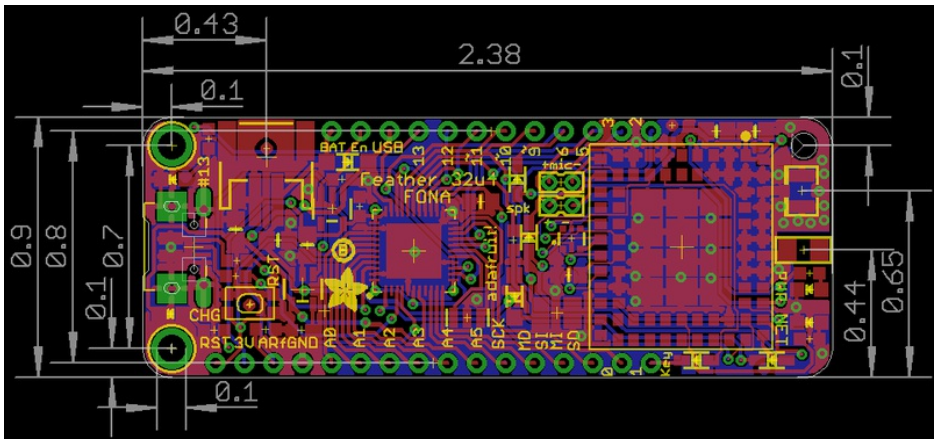
The 'classic' Feather and Wing size is **0.9" x 2.0"** with **0.1"** holes at each corner.

There is one 16-pin breakout strip on the bottom side, centered 1.0" from the left edge. There is one 12-pin breakout strip on the top side, 1.2" from the left side. The spacing between the two strips is 0.8" **Don't change the GPIO spacing or location, or you will not maintain compatibility with Wings!**

This Proto Featherwing shows off those dimensions:



If you need more space, you can, of course, go double-sided **or** you can go longer. For example, the 32u4 FONA Feather has the same width but is longer:

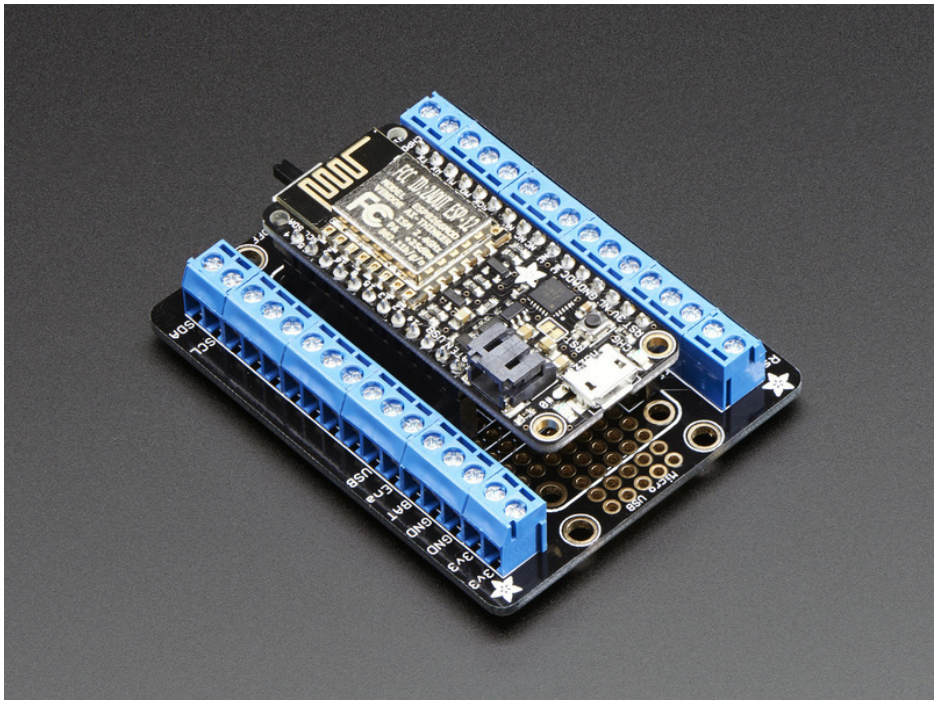


If you are making a **Feather main-board**, we don't recommend going any wider, or adding space on the 'left side'

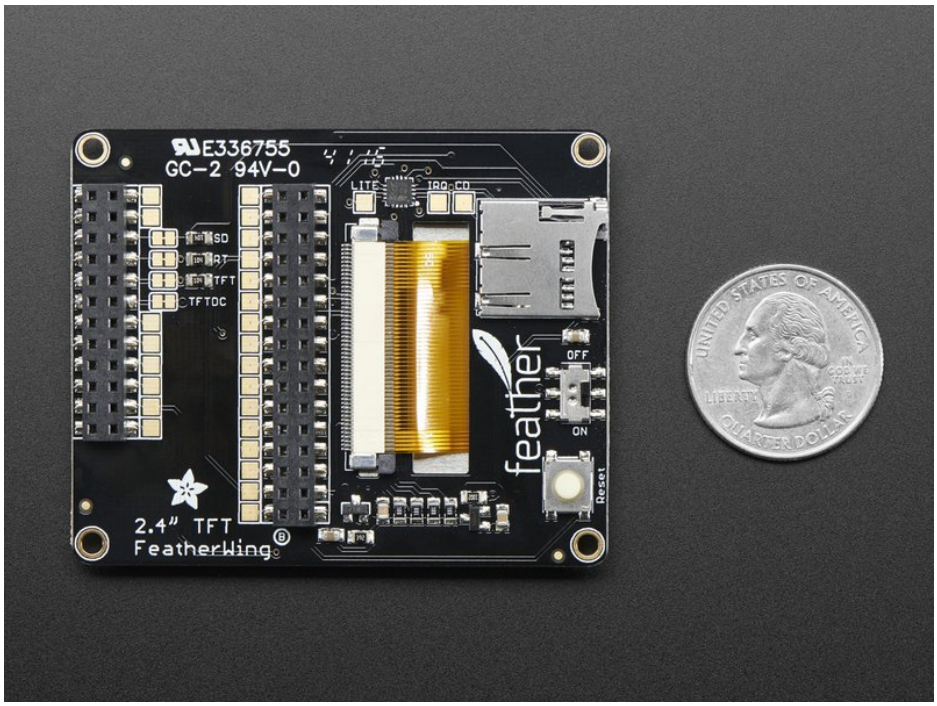
where the USB port is. If you do go wider, or add space, make the USB side (the left two holes, and the breakout pins with-respect-to bottom left corner) the same.

If you *do* go wider, it wont fit nicely in a breakout board.

For **Feather Wings**, you don't have to keep the same size. You can have it be a 'bottom' Wing like this terminal block Wing, where the Feather pops on top:



Or you can do what we did with the TFT FeatherWing where the board plugs into the back.



Note that even with these larger wings, you can use stacking headers to add more wings on top.

Pins & GPIO

One of the goals of Feather was to unify the pinouts so that we only have to make *one* OLED display and have it work with all the mainboards.

If you maintain compatibility you can expect to be able to work with any current or future Wings! (Note, however, we only test our own boards for guaranteed compatibility)

Power

All Feathers and Featherwings use **3.3V logic** - so do not make anything that *requires* 5.0V logic. **You must accept and emit 3.3V logic on any Feather/Wing** (except for Analog inputs)

The power pins on all Feather mainboards must be the same:

VBUS

This is the power that comes from the USB port. It should be 4.5-5.5V or so, 500mA. You can put a fuse on this pin if you like, but it is not required. This pin can be at approx the same voltage as VBAT if the USB power is removed (some of our older Feathers did this), or it can disconnect completely (our newer Feathers including the ESP32 do this with a transistor switch). FeatherWings can use this for powering larger items like servos or lots of LEDs, but they need to cope/understand that it can go away when the Feather is unplugged from USB

VBAT

This is the power that comes from the 'optional' LiPoly battery. It should be 3.0-3.7V or so, with variable current capability.

If you are making a mainboard Feather you must have onboard LiPoly charging (its a core expectation) and the LiPo connection **must be a JST 2-PH** that matches Adafruit batteries. The connector must be in the same location, 0.425" from the left edge

Also, you must have some charge indication LED, we strongly recommend sticking a yellow CHG LED below the microUSB connector.

We tend to require no smaller than a 250mAh battery for WiFi/Radio Feathers. On the Cellular Feathers, its reasonable to require 750mAh or higher, and have the charge rate be 500mA. We don't recommend setting the charge rate below 100mA or above 500mA but as long as its well documented why you chose it, its fine.

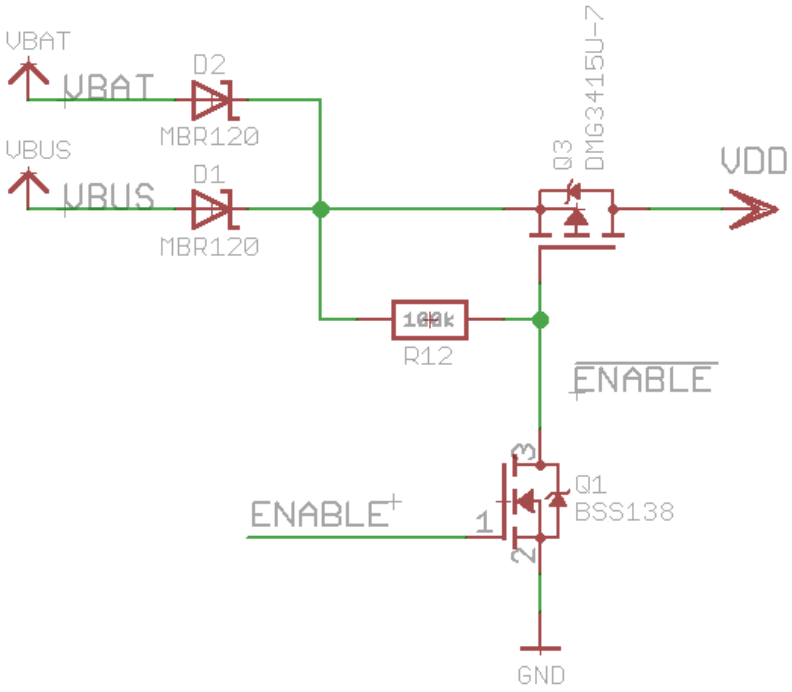
We strongly recommend you design your Feather and Wings to not require a LiPoly plugged in. For cellular, a LiPo is required but that is the one exception we've made so far!

FeatherWings can use this pin for powering larger items like servos or lots of LEDs, but they need to cope/understand that it can go away if the LiPo dies or if it is not unplugged.

VBAT & VUSB Usage on Wings

We recommend that if your Wing *does* need high power, to use 2 x 1A Schottky diodes to give higher-of-either for your power input.

We also recommend that you use a simple invert+switch to disable the power source when **EN**able is low:



3.3V Output

The 3.3V pin is power **from** Feathers **to** Wings. You must have a regulator or buck converter on the Feather mainboard to take VUSB/VBAT to 3.3V. It is OK to have the 3.3V drop down to 3.0V when the battery is dead. We like the AP2112K-3.3 regulator, it's ultra low dropout, a good price, and has up to 500mA current output. If you pick something else, we recommend that after powering the Feather, you have at least 100mA remaining in the budget for Wings.

ENable Input

The **EN** pin is an *input* to your Feather Mainboard and **must be used to depower the 3.3V output**. We tend to tie it to the Enable pin on the 3.3V regulator, with a 100K pullup. It does not disable VBAT/VUSB, see above for a circuit that will let you do that. It is OK if you decide to have EN de-power USB/BAT as well, but not required.

Bus Pins

To make it easy to create add-ons, we have fixed inter-chip bus pins:

- **RX & TX** - these are UART pins. If your mainboard has a spare hardware UART, put these here. If your board uses the sole UART for bootloader/debug (e.g. ESP8266) then you can put those here as well. If designing a Feather, try to not use these as they are not always available
- **SDA & SCL** - this should be your main I2C bus. I2C pullups are put on the Wings, not on the mainboard. These are 3.3V logic, we strongly recommend avoiding clock-stretch I2C on Featherwings. Repeated start is OK! This is our recommended interface for sensors and Wings, try to stick to it.
- **SCK/MOSI/MISO** - this should be your main SPI bus. 3.3V logic. If you share these with devices on the main featherwing, keep the CS pin from being exposed so that you don't have SPI bus contentions.
- **I2S** - we don't fix the I2S pins! I2S is not as common and often are very restricted. We don't have any guarantees on where the I2S pins may lay.

Analog Pins

There are 6 reserved spots for Analog pins between the power pins and SPI pins. You do not have to have the analog pins in order (e.g. A0 first, then A1) but it is considered in good taste to do so. We do our best to put any DAC pins on the first two pins (e.g. Feather M0 has the DAC on A0, ESP32 has the DACs on A0 and A1)

The ESP8266 only has one analog pin and its 1.0V max, so be aware that while every other Feather mainboard has 6 analog-ins, if you want your wing to work with the ESP8266 there are restrictions. We tend not to use the analog inputs on Wings for these reasons.

While we dont *require* all 6 pins to be analog inputs, it would be unusual if they were not. So please do your best to keep those pins analog!

General Purpose Pins

Between BAT and I2C pins there are 7 GPIO pins you can break out. These are every day 3.3V GPIO pins. The vast majority of Feather boards have these pins available and with customizable pullups.

However, the ESP8266 is the one constrained board and shares 3 of the GPIO with SPI and the other 4 have some unusual pull up/down requirements. So if you make a Wing that uses these GPIO, please test it with the ESP8266 Feather as it is the most challenging, or make it clear that the ESP8266 Feather is not compatible.

We don't expect all FeatherWings that use these GPIO to be stackable with other GPIO-using Wings (only I2C is really stackable) so feel free to use these however you like.

Other Pins

- **AREF** - this is the Analog Reference if there is one for the chip. If there is not, keep this pin not-connected. We recommend Wings don't require this pin as not all chips have external ARef's.
- **RESET** - this is the main feather board reset line. **It must be active low** - that is you tie it to ground to reset the main Feather. Nearly all our Wings have a reset button that can be used. **You must have a reset tactile button on your Feather.**
  A reset button on Wings is not *required* but is strongly recommended as the Wing usually covers up the button!
- **'Free' Pin** - This is the pin to the right of TX. You can use it for an extra GPIO or if you have some onboard module that has a useful breakout. Sometimes we tie it to ground. Do whatever you like! FeatherWings should not require or use this pin unless there's some really good reason.

Other Requirements

USB

USB is used for debug, upload and battery charge. We use micro B USB but you can also use USB C if you like! USB B wont fit (too chunky) and Mini B is deprecated so please stay away from those two.

Feather mainboards **must have USB debug upload with a micro USB connector in the center left of the board.** If your mainboard has native USB debug/upload, use that! If you do not, use a USB-serial converter chip such as CP2104, FT23x, etc.

You do not need to have RX/TX USB activity LEDs, but they're nice if you can!

Bootloader / User LED

There must be at least one USER LED, we recommend it also indicate when the bootloader is active. **Put this LED above the microUSB connector**. We recommend red but not required. You can share this with one of the general

purpose 7 IO pins, but if you can have it be separate and not-exposed, that's best!

Not Required!

- JTAG, ISP or SWD connector - if you can fit one, great! But it is not required if you have a bootloader.
- Bootloader - if your board for some reason cannot come with a bootloader (because, say it is an FPGA board) just make it super clear. For anything else, a USB bootloader is very, very strongly recommended. UF2 file update support is required for CircuitPython-compatible boards.
- Power LED - we don't include one in general, but a power-good LED is fine to add
- Additional Breakouts - if the 12+16 pins are not enough, you can add more breakout strips if you have space - but note that FeatherWings may not use them

CircuitPython

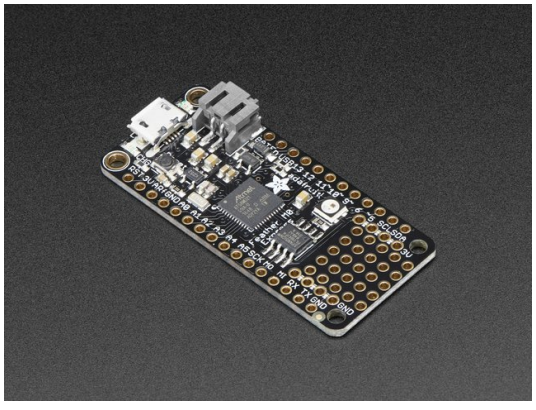
All of our Feathers support Arduino and C/C++ using a gcc toolchain. But not everyone wants to write C/C++ code. Thanks to the bigger and faster chips available, it is now possible to run a *full Python interpreter directly in the microcontroller itself!*

mind blown

The original version of this embedded Python is called MicroPython, written a feat of magic by Damien George (https://adafru.it/pF5). Please help support him by purchasing from his shop (https://adafru.it/wf9)! We also stock MicroPython boards, books and badges in the Adafruit shop (https://adafru.it/wfa) all of which directly help Damien's efforts.

CircuitPython is a variant of MicroPython which is designed specifically for beginners, and use the chipsets on Adafruit boards, like the SAMD21 and SAMD51. They are 98% the same and use the same interpreter core, just some of the helper libraries vary in naming.

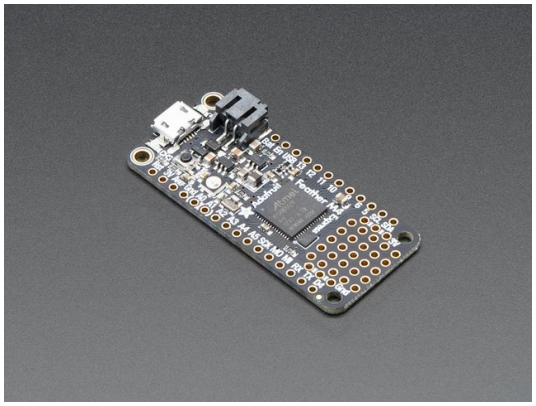
Feathers designed for CircuitPython



Adafruit Feather M0 Express - Designed for CircuitPython

$19.95
IN STOCK

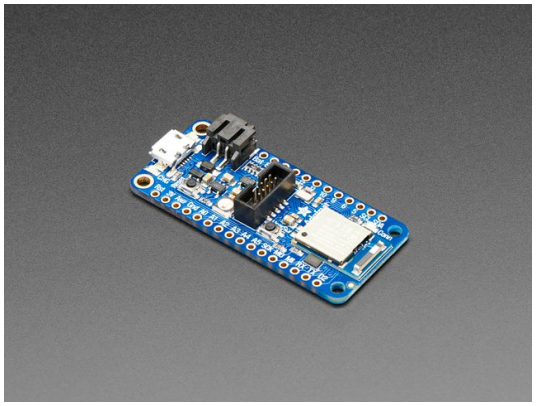
ADD TO CART



Adafruit Feather M4 Express - Featuring ATSAMD51

$22.95
OUT OF STOCK

OUT OF STOCK



Adafruit Feather nRF52840 Express

$24.95
IN STOCK

ADD TO CART

Your browser does not support the video tag.

Adafruit HalloWing M0 Express

$34.95
IN STOCK

ADD TO CART

The Feather M0 Express and M4 express are the first in a line of products specifically designed for CircuitPython (https://adafru.it/tBa). The SAMD21 and SAMD51 microcontrollers are inexpensive while still being full featured with native USB, multi serial (I2C and SPI) engines called SERCOMs and a 10 bit DAC. Its 256k on board flash is supplemented with an external flash chip that gives a huge two megabytes for your code and all of our CircuitPython libraries (https://adafru.it/uap). The status NeoPixel gives you information on the state of CircuitPython even when its unplugged from a computer. Its perfect for rapid prototyping of sensor driven projects.

The Feather nRF52840 Express is a newer board that leverages both a Cortex M4F processor and a Bluetooth radio.

Contrary to its name, the Adafruit HalloWing M0 Express is a Feather (not a FeatherWIng) in a modified form factor with an LCD display. The closeness of HalloWing and Halloween was too good to pass up.

Pros

- UF2 bootloader for easy drag and drop flashing
- 2MB SPI Flash for storing all your code, libraries and data files
- Status NeoPixel for debugging in a pinch
- Built-in USB support enabling CIRCUITPY drive, USB HID (mouse and keyboard) and easy serial

Cons

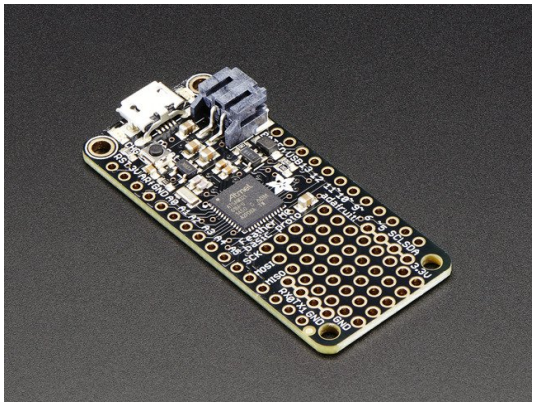
- No built in radio on the SAMD boards. Bluetooth for the nRF52840. Additional radios are available on FeatherWings.
- Not compatible with MicroPython, only CircuitPython
- M0: CircuitPython interpreter uses half the 32KiB RAM — only 16KiB for the user

CircuitPython-Compatible Feathers

These Feathers weren't *designed* for CircuitPython, but they work just fine!

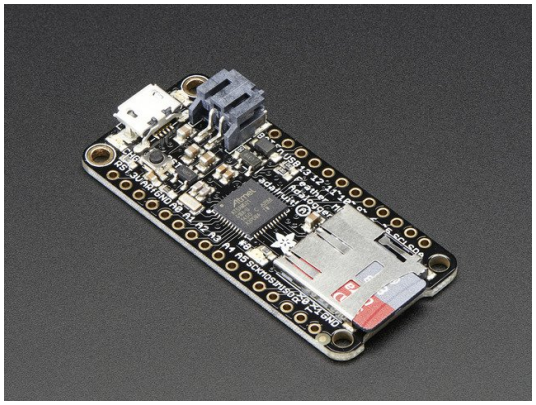
The Huzzah does not have native USB so you don't get the nice "USB drive with source code on it". The non-Express M0's don't have a 2MB SPI Flash so the USB drive you get is quite small, and gets wiped if you update the firmware.

So if you really want to have an easy CircuitPython experience, we suggest the M0 Express, above!



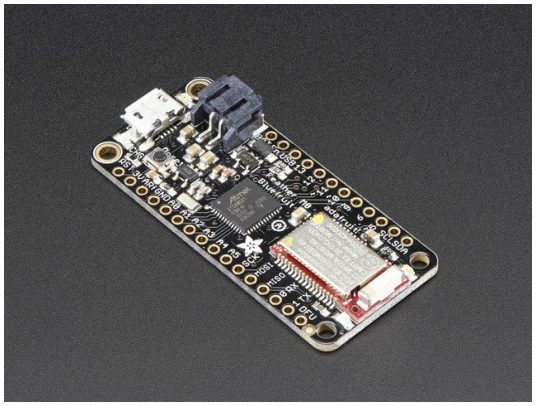
Adafruit Feather M0 Basic Proto - ATSAMD21 Cortex M0

$19.95
IN STOCK
ADD TO CART



Adafruit Feather M0 Adalogger

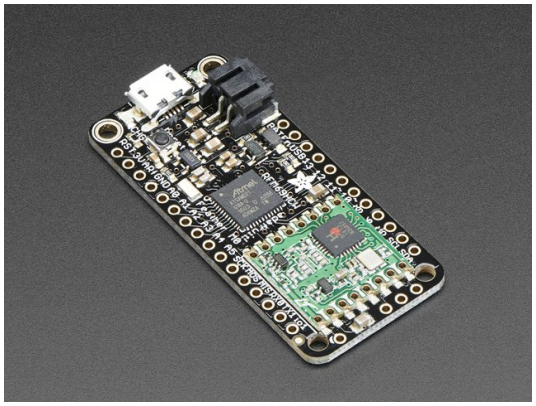
$19.95
IN STOCK
ADD TO CART



Adafruit Feather M0 Bluefruit LE

$29.95
IN STOCK

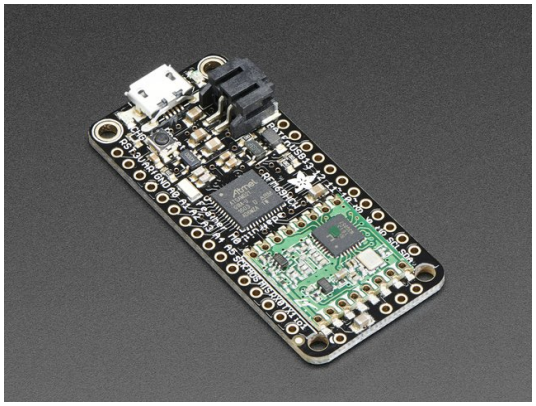
ADD TO CART



Adafruit Feather M0 RFM69HCW Packet Radio - 433MHz

$24.95
IN STOCK

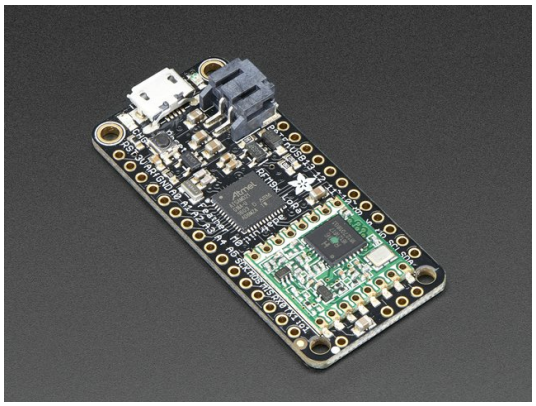
ADD TO CART



Adafruit Feather M0 RFM69HCW Packet Radio - 868 or 915 MHz

$24.95
IN STOCK

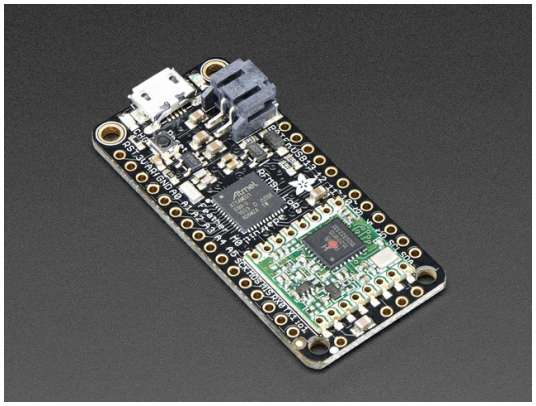
ADD TO CART



Adafruit Feather M0 with RFM95 LoRa Radio - 900MHz

$34.95
OUT OF STOCK

OUT OF STOCK



Adafruit Feather M0 RFM96 LoRa Radio - 433MHz

$34.95
IN STOCK

ADD TO CART

Prior to CircuitPython, the SAMD21 M0 microcontroller wasn't supported by MicroPython. However, it was supported by Arduino and that lead to a wide variety of M0 based feathers. Now, all of these can run CircuitPython but are more limited in the functionality available. They run the same speed as the Feather M0 Express but are missing the extra space for code on the SPI flash which also means that there is less room internally for functionality in the CircuitPython core.

So, if you already have one and want to try CircuitPython, then you can! They can do smaller code tasks just fine. Just beware that it won't be quite as easy as the Feather M0 Express.

Also note that the extra functionality on these boards, such as the SD card, radios and Bluetooth modules, is not yet supported in CircuitPython. Some of them will require additional core CircuitPython work while others will only require a new library or two.

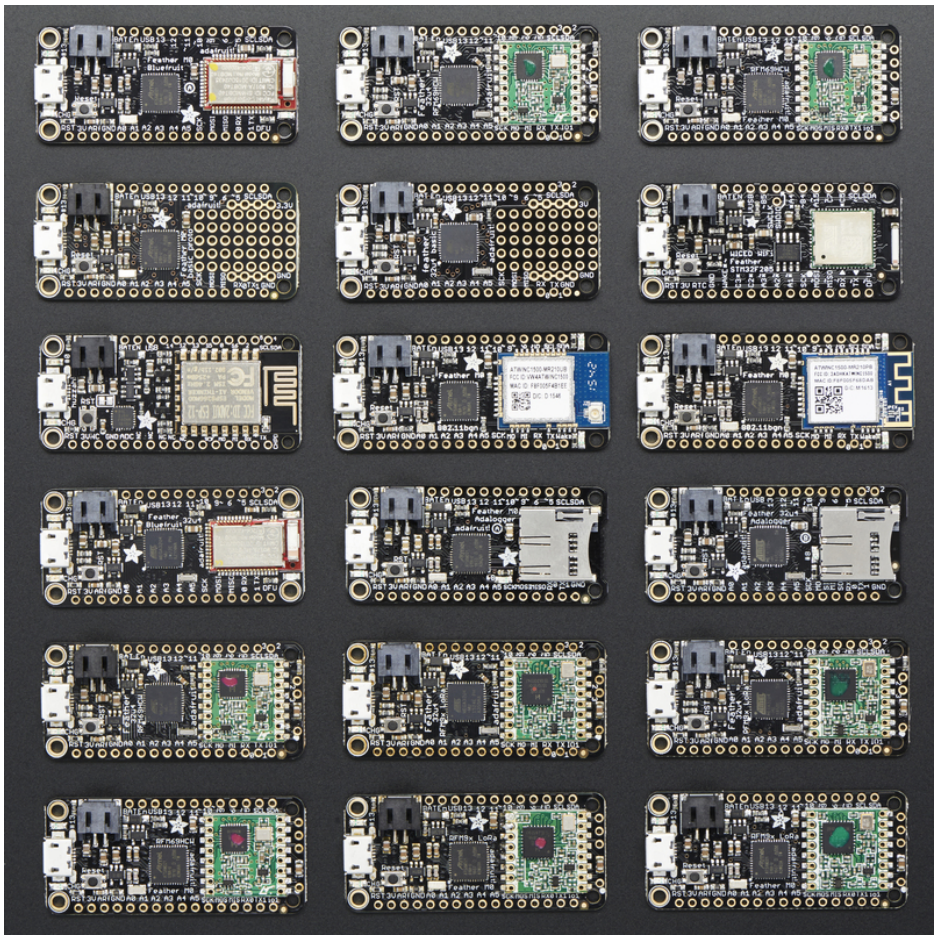
Pros

- Additional functionality on-board
- Built in USB support for CircuitPython drive
- If you already have one, you can try CircuitPython

Cons

- Very small file storage 64KiB max
- No UF2 bootloader
- No CircuitPython libraries for extra functionality
- CircuitPython version is reduced functionality to share flash with file storage

Feathers



Feathers are the *main/mother boards* of the Feather platform.

You always need **one Feather** and can add **zero or more FeatherWings** to increase the capabilites.

For example:

- Feather Huzzah ESP8266 + Music Maker FeatherWing = MP3 streaming player
- Feather 32u4 Bluefruit + NeoPixel FeatherWing = Bluetooth LE controllable RGB lamp
- Feather 32u4 FONA + GPS FeatherWing = Cellular GPS locator with SMS control
- Feather nRF52 Bluefruit + Motor FeatherWing + Robot Chassis = Phone-controlled robot rover
- etc!

All Feathers...

- All Feathers have a microcontroller that can be programmed by Arduino IDE
- All Feathers are 3.3V logic
- All Feathers come with a MicroUSB connector that is used for data and power
- All Feathers have a bootloader, and do not require a separate programmer device
- All Feathers have USB Serial for debug (sometimes it is part of a native USB interface, sometimes via a USB-serial adapter chip)
- All Feathers are 0.9" wide, and fit in a breadboard with 1 row available on one side, and two rows on opposite

side.

- Almost all are 2.0" long (M0 ATWINC Feathers, FONA Feathers, and Hallowing M0 Express are the exceptions)
- All Feathers have the same two mounting holes near the USB port
- All Feathers have Power, I2C, SPI, UART pins in the same location. Remaining pins try to be as similar as possible (but they may have different pin names/numbers)
- All Feathers can run from USB or LiPo battery, and have a LiPo charger built-in

Most Feathers...

- Most of the Feathers have an **En**able pin that will allow shutdown via logic level (Teensy and FONA Feathers are the exceptions)
- Most of the Feathers have a native USB connection (nRF52832 and ESP8266 are the exceptions)

Some Feathers...

- Some Feathers have a wireless module built in. Those that do not most likely can have an added FeatherWing to provide a radio.
- Only Feathers with "Express" in their name are fully CircuitPython Compatible.
- Feathers with a SAMD21 "M0" processor but not an "Express" board may run CircuitPython but must share the onboard flash with CircuitPython, restricting the space available.

No Feathers

- No feathers are guaranteed to be 5V-input compliant. Use **3.3V logic** only please!

Basic Feathers

Basic feathers are..basic. But they're not to be ignored! Their simplicity makes them a great base for Feather projects, when you don't need a wireless network, or built-in datalogging.

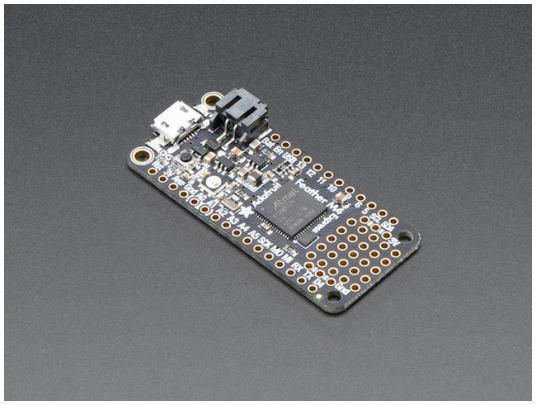
These were the First Feathers, as it were. They form the basis of later Feathers, so if you start here you can upgrade!

There's four chipsets available:

- Atmel (now Microchip) ATmega328P
- Atmel (Microchip) ATmega32u4
- Atmel (Microchip) ATSAMD21 Cortex M0
- Atmel (Microchip) ATSAMD51 Cortex M4
- Freescale MK20DX256 - A.K.A Teensy 3.2, via an adapter

Note that there are ATSAMD21, ATSAMD51, and nRF52840 Express Feathers with extra flash onboard for CircuitPython use listed in the CircuitPython page (https://adafru.it/DO1).

ARM ATSAMD51



Adafruit Feather M4 Express - Featuring ATSAMD51

$22.95
OUT OF STOCK

OUT OF STOCK

Anything but basic, the Adafruit Feather M4 is the top of the line Feather for heavy lift.

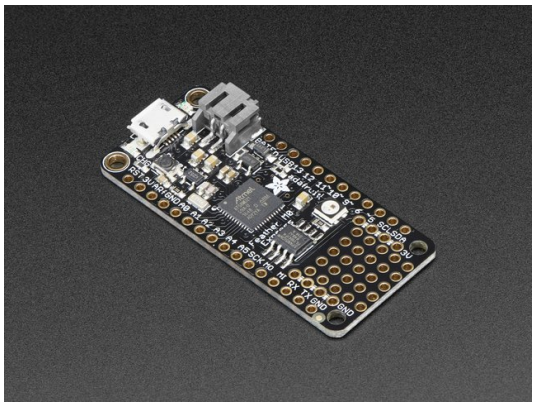
It is powered by our new favorite chip, the **ATSAMD51J19** - containing a **120 MHz Cortex M4** with floating point support, **512KB Flash and 192KB RAM**. Your code will zig and zag and zoom, and with a bunch of extra peripherals for support, this will for sure be your favorite new chipset.

The most exciting part of the Feather M4 is that while you can use it with the Arduino IDE - and it's bonkers fast when you do, we are shipping it with **CircuitPython** on board. When you plug it in, it will show up as a very small disk drive with main.py on it. Edit main.py with your favorite text editor to build your project using Python, the most popular programming language. No installs, IDE or compiler needed, so you can use it on any computer, even ChromeBooks or computers you can't install software on. When you're done, unplug the Feather and your code will go with you.

ARM ATSAMD21

The ATSAMD21 M0 is a 32-bit Cortex M0, with a ton more flash and RAM: 256KB and 32KB. It is also supported in Arduino but is a little newer (circa ~2015) and so does not have as many years of projects and example code already. It's also a completely different chipset than the AVR, so low level code will not work and must be re-written.

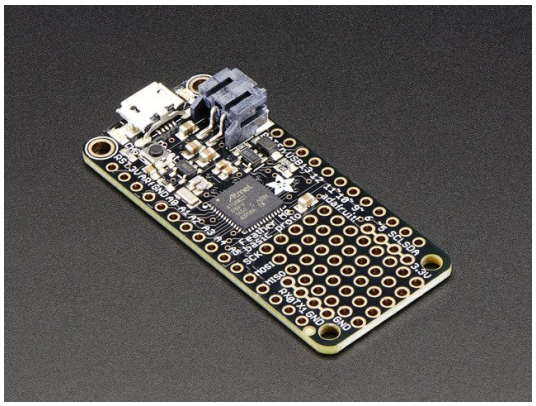
That said, its about the same price and you get a ton more speed (48 MHz and 32-bit processing), and peripherals. For example you can easily create multiple hardware UARTs or I2C ports, which is impossible on the AVR. Other stuff like I2S audio, 12-bit analog input, and true analog output (DAC) is only available on this chip. Especially if you're running low on Flash, RAM, or analog inputs, this is a nice upgrade.



Adafruit Feather M0 Express - Designed for CircuitPython

$19.95
IN STOCK
ADD TO CART

The Feather M0 Express has add a **Mini NeoPixel**, **2 MB SPI Flash** storage and a little prototyping space. Perfect for using with CircuitPython but also runs great with Arduino.

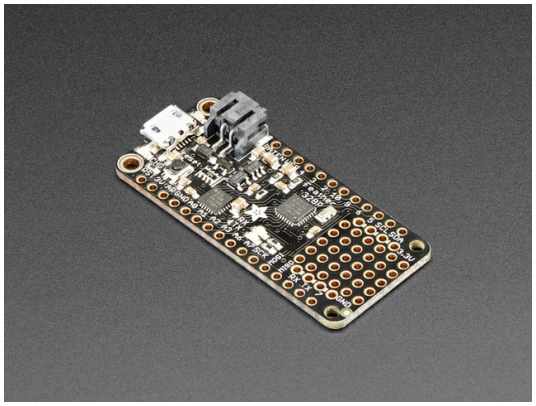


Adafruit Feather M0 Basic Proto - ATSAMD21 Cortex M0

$19.95
IN STOCK
ADD TO CART

The Feather M0 Basic is a speedy chip mostly designed for Arduino programming. It can run CircuitPython but the board lacks the extra off-chip flash for extra storage space.

AVR ATmega 328P



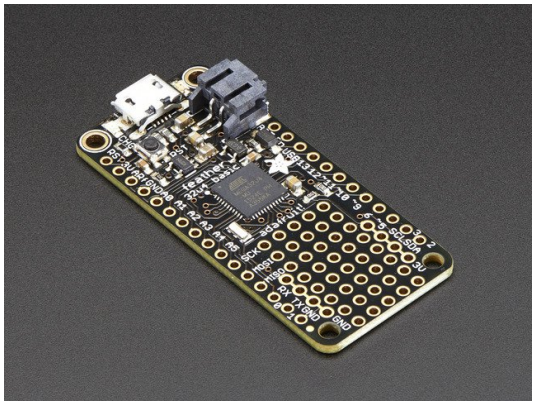
Adafruit Feather 328P - Atmega328P 3.3V @ 8 MHz

$12.50
OUT OF STOCK

OUT OF STOCK

The ATmega328P is nearly identical to the ATmega328 processor on the Arduino Uno and related boards. The Arduino core is rock solid. It still carries the same specs: 32 KB Flash and 2 KB of RAM. It runs at 8 MHz and uses a CP2104, a USB serial converter for USB bootloading and serial port debugging. There are 19 GPIO pins + 2 analog-in-only pins available on this tiny board.

AVR ATmega32u4



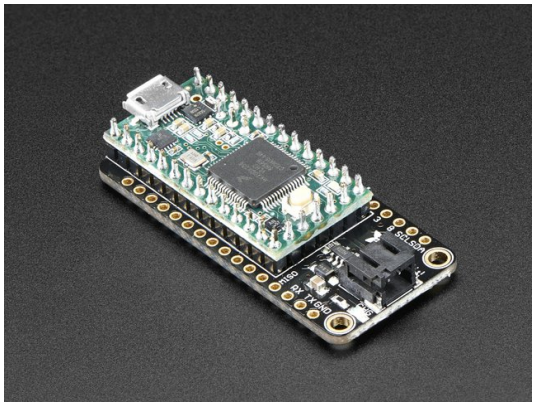
Adafruit Feather 32u4 Basic Proto

$19.95
IN STOCK

ADD TO CART

ATmega32u4 is a well known 8-bit AVR processor, and may be more familiar to people who use the similar Arduino '328s. It's tried-and-true. The Arduino core for it was first developed in 2012 and so it's quite stable and well-known. Many low-level ATmega328 examples will work out of the box on the 32u4 (but not all!). It has 32KB of Flash and 2.5KB of RAM. It runs at 8 MHz and has native USB support

Teensy 3.2 Adapter



Teensy 3.x Feather Adapter

$5.95
IN STOCK

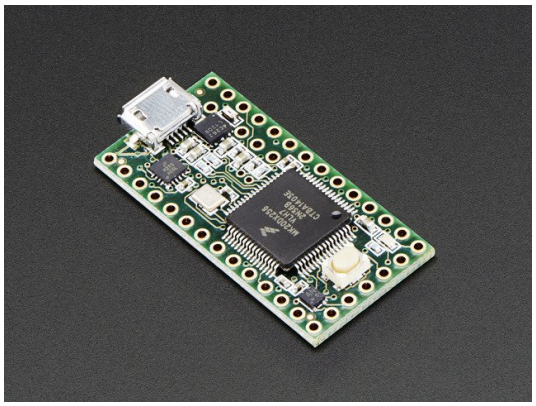
ADD TO CART

It turns out the Teensy 3.2 fits right between the headers of a Feather, so we thought - hey lets make this into an adapter!
You get the awesome power, speed, and projects of the Teensy 3, with the wings of a Feather.

It's a great way to take advantage of the Feather ecosystem. You dont get *everything* - for example the **EN** pin isn't connected. But we've tested our FeatherWings (https://adafru.it/vby) so far and all are drop-in compatible.

With the space left over, we even added in a 500mA LiPoly charger that automatically charges over USB and will switch over to the LiPo when USB is unplugged. There's also a 100K resistor divider for monitoring the battery voltage connected to A7

Teensy not included, so be sure to pick one up, too!



Teensy 3.2 + header

$19.95
IN STOCK

ADD TO CART

WiFi Feathers

Around 2012, low cost WiFi embedded modules started appearing to makers. At first they were $20 each but the introduction of the TI CC3000 broke the $10/ea barrier (in large qty) and so embedded boards with WiFi started coming on the market. Then in 2014, Espressif released a $2 WiFi-inclusive microcontroller called the ESP8266 and competition heated up. Now there's a lot of WiFi options available for makers to create IoT projects!

About WiFi

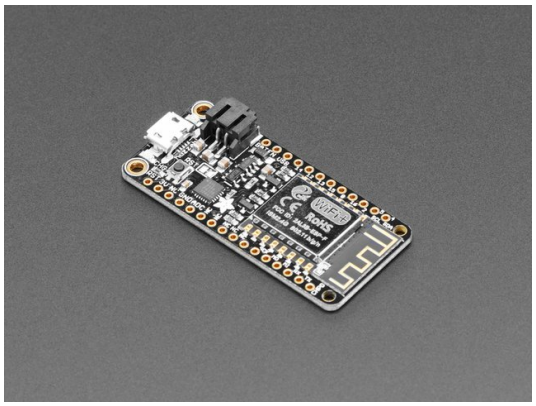
WiFi is just about everywhere but there's a few things to watch out for:

- Compared to Bluetooth, WiFi uses a ton of power, you can easily end up using 250mA during transmit and receive/listening. Try to use low power modes to reduce that if possible
- Compared to ZigBee, BTLE or LoRa/Packet radio, WiFi can transfer a lot more data a lot faster! You can easily stream compressed audio over WiFi.
- Compared to Ethernet, you may have connectivity problems, just like you do when your laptop has poor reception
- Compared to Cellular, WiFi tends to need a base-station nearby, so you can't use it in the middle of nowhere unless you have a WiFi hotspot
- Compared to just about any other wireless protocol, WiFi is strongly supported by every computer/tablet/phone, has strong encryption built-in, and can reach anywhere around the world!
- Like LoRa & packetized radio, you *can* create a WiFi ad-hoc network but this usually requires a little extra effort

All of our current WiFi Feather options support WPA PSK passcoding and SSL encryption. The full capabilities of SSL may vary, and you may have to do a little extra work for adding the SSL certificate to the module for true SSL checking.

WiFi Feather Options

We have a few WiFi Feathers available



Adafruit Feather HUZZAH with ESP8266 - Loose Headers

$16.95
IN STOCK

ADD TO CART

The Feather Huzzah ESP8266 is a perennial favorite. It features an FCC/CE certified ESP-12 module that contains an ESP8266 chip, 4 MB of SPI flash and antenna.

Pros:

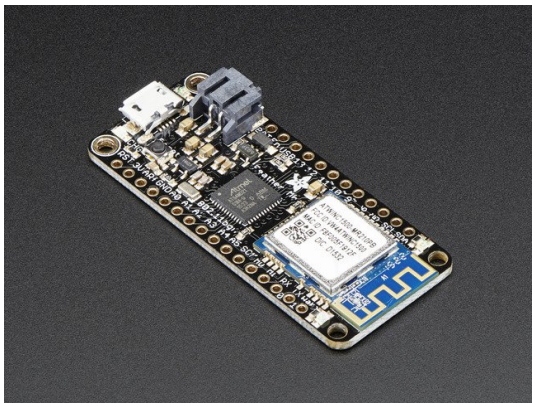
- The ESP8266 is super popular, has tons of projects, tutorials, guides
- Arduino support is very good, with a community/Espressif supported build that has been updated and maintained
- Can also be used with MicroPython or Lua

- Fast and reliable WiFi connectivity with SSL support
- Speedy chip at 80 MHz
- Very affordable, lower cost than other WiFi chipsets

Cons:

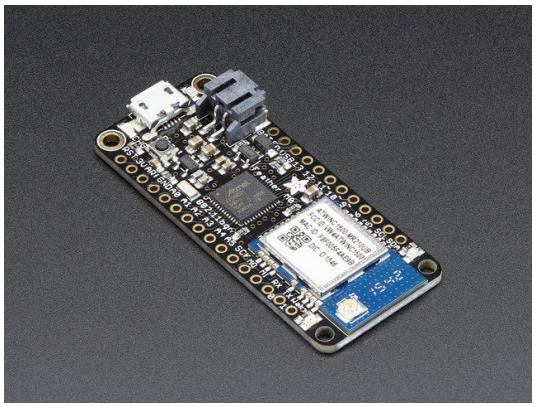
- High power draw, no easy-to-use sleep modes
- Single core design has a real time component that is not documented but required cycle time, so Arduino code needs to constantly yield() or the RTOS component will reset/crash
- Not a lot of GPIO pins, many have special functions so you can't use them for any purpose
- Only one analog input pin, 1.0V max
- No real datasheet or in-depth documentation. Most information is community-sourced or 'word-of-mouth'. Tech support is minimal.

For the most part, the low cost of the ESP8266 has been enough incentive for people to overlook the drawbacks of the chip and figure out how to create projects with what they've got.



Adafruit Feather M0 WiFi - ATSAMD21 + ATWINC1500

$34.95
IN STOCK
ADD TO CART



Adafruit Feather M0 WiFi with uFL - ATSAMD21 + ATWINC1500

$34.95
IN STOCK
ADD TO CART

The Feather M0 + ATWINC1500 is a pairing of chips: there's a main processor (the Feather M0 part) and the wifi processor module (the ATWINC1500 part). As such, these Feathers are more expensive than all-in-one WiFi solutions. But, as a positive, they have a really powerful and well-documented main processor that runs separately from WiFi which can give you more control.

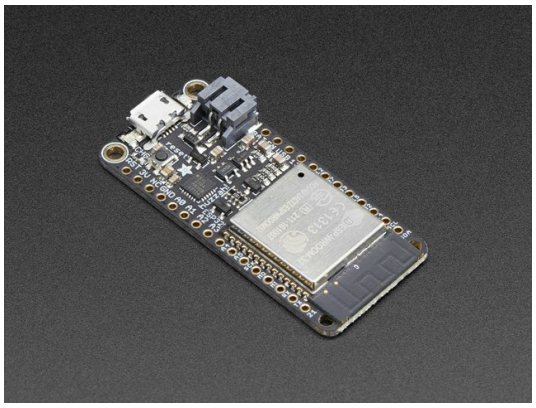
Pros:

- Main chip is ATSAMD21G18, which has solid official Arduino support
- Tons of GPIO and peripherals: analog inputs, I2S audio, DMA and even analog output support

- Proper sleep modes, can shut down WiFi module completely
- ATWINC1500 has some low-power mode support
- Fast and reliable WiFi
- Good SSL support, including SSL certificate uploading for fingerprinting
- Main processor is well-known and documented Cortex M0+ chip. WINC1500 has official support from Atmel/Microchip.
- Can use external antenna with the uFL version

Cons:

- More expensive than single-chip solutions
- Firmware on the module is opaque, no way to really debug or analyze beyond the firmware provided
- Not as popular as ESP8266 so fewer projects published
- No current MicroPython support for the ATWINC1500



Adafruit HUZZAH32 – ESP32 Feather Board

$19.95
IN STOCK

ADD TO CART

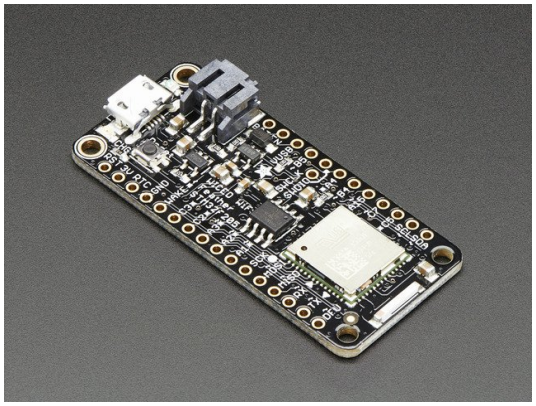
The ESP32 is the 'big sister' the the ESP8266. It has two cores, so that one can take care of the wireless management and data while the other one does processing

Pros:

- High speed dual chipset has *tons* of processing capabilites
- Lots of GPIO, analog inputs, two analog outputs, peripherals a-plenty!
- WiFi, Bluetooth LE and Bluetooth Classic all in one chipset, so you can do a wide range of IoT
- Plenty of example code support from Espressif, and they've hired skilled community members to write code, libraries, Arduino core support, and projects
- Low power sleep support
- MicroPython support

Cons:

- Slow roll-out means not as many projects for this chipset yet, but it's gaining popularity
- No detailed documentation about peripherals. Support is offered as example code.
- As of mid-2017, core WiFi and BTLE functionality is supported but some capabilities like classic BT or dual-wireless projects are still in-the-works



Adafruit WICED WiFi Feather - STM32F205 with Cypress WICED WiFi

$34.95
OUT OF STOCK

OUT OF STOCK

The WICED Feather builds on a Broadcom (now Cypress) module, the same used in the Particle Photon. This chip combines an STM32 and BCM radio. Thanks to Thach's amazing software stack, we've taken a very complex and powerful chipset pair and made it trivially easy to get working within Arduino.

Pros:

- Powerful ARM Cortex M3 MCU (STM32F205) running at 120MHz
- Works directly from the Arduino IDE, generating native ARM code that runs directly on the target MCU.
- Plenty of GPIO and powerful peripherals, including native USB support with a custom USB DFU bootloader for programming from the Arduino IDE or from the command line.
- Fast throughput due to the efficient WiFi stack and fast multi-lane connection between the MCU and the Broadcom radio.
- Advanced features for various encryption modes, including TLS 1.2 (required for Amazon Web Services, etc.).

Cons:

- The WiFi stack and security libs are black box closed source solutions, meaning we have little influence over bug fixes and getting changes into the code. We're largely at the mercy of Broadcom (now Cypress) for any bug issues, and like any very complex stack there are known bugs in the security and wireless layers.
- No access to the low level source code since it is under very strict NDA, and we had to bend over backward to implement a layer that sits on top of the NDA code so that we have something we can expose.

Bluetooth Feathers

About Bluetooth

Bluetooth is a 2.4GHz wireless protocol that is popular for short range - about 10 meters max. It's very low power and simpler than WiFi. That low power and simplicity has made it the most popular point-to-point wireless transport - you now get a Bluetooth transceiver in every phone, tablet and laptop.

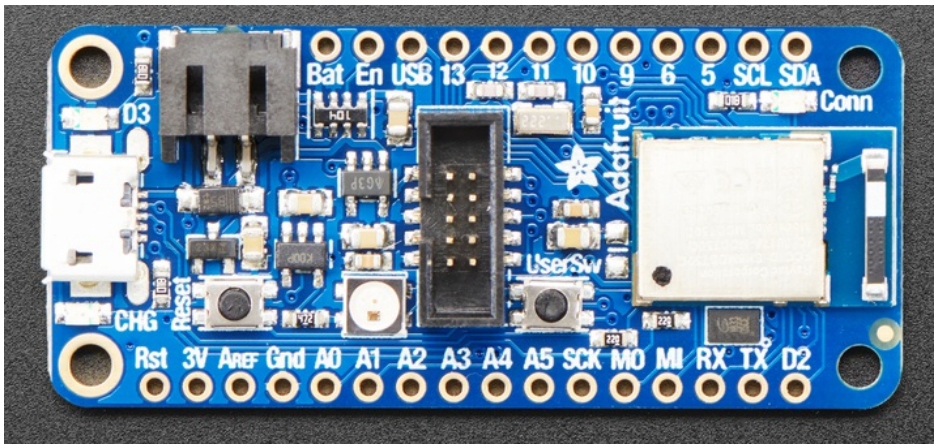
There are, annoyingly, two *flavors* of Bluetooth: BT Classic and BT Low Energy (introduced as part of the Bluetooth 4.0 standard). And, annoyingly, despite the similar names, they are completely different and not cross-compatible.

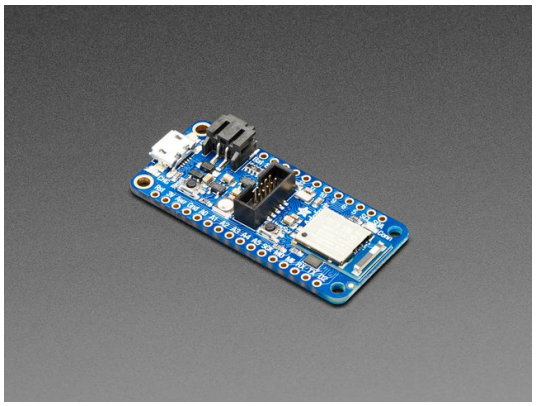
Bluetooth classic is used for unstructured serial data transfer (sometimes called SPP), keyboards and mice, and audio such as BT speakers or BT headsets

Bluetooth LE is used for structured data (organized by 'Services', which are made up of individual 'Characteristics'), beacons, keyboards and mice, MIDI, but not audio. It is also the only method exposed to the user in iOS.

Basically the thing to watch for is if you want to communicate with SPP devices or BT audio, you can't use LE. If you want to work with iOS, you can't use Classic. Keyboards/mice can be either.

Feather nRF52840 Express (nRF52840)





Adafruit Feather nRF52840 Express

$24.95
IN STOCK

ADD TO CART

The Feather nRF52840 builds on the nRF52832, the core and peripherals are very similar. It adds 2x the FLASH and 4x the SRAM and also comes with a native USB client peripheral. Like the nRF52832 Faether, it has only *one* chip on it

- and that chip is both the processor you program and also the **Bluetooth Low Energy radio**. What's nice about this is you can do more powerful stuff, and faster too, because you don't have to manage two chips. It's also lower price and lower power since there's only one processor, and easier to put into sleep modes.

The native USB means that its great for Arduino or for CircuitPython support - no other BLE chipsets have USB!

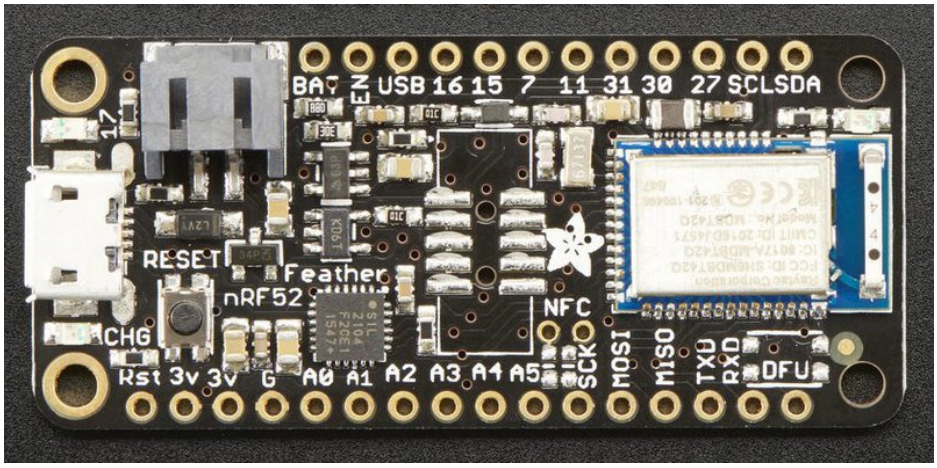
Pros:

- Code runs natively on the nRF52840 MCU, so it's fast, efficient and less expensive than the two MCU based nRF51 boards
- Powerful ARM Cortex M4F processor with 1MB flash and 256KB SRAM
- Native USB! Can act like a keyboard, mouse, MIDI or disk drive. Bootloader can be mass-storage based.
- Supported by CircuitPython
- Hardware UART pins are available for communicating to other peripherals (unlike the nRF52832 which needed them for the bootloader chip)
- Better low power potential due to being a single chip solution
- Because everything runs natively, a lot of complexity can be hidden behind simple helper classes, making things like ANCS (https://adafru.it/wfj) possible and manageable compared to the nRF51.
- Supports both Central and Peripheral mode, Central mode meaning it can behave like a phone and initiate connections to other peripherals, although Central SW support is in the very early stages at this point
- Much more room to grow due to the MCU capabilities and flash/SRAM size compared to the earlier nRF51 or nRF52, and more advanced on board peripherals.

Cons:

- A new chipset that is not an ATmega328 or ATmega32u4, so some older low-level Arduino libraries and techniques are not compatible. This doesn't happen *a lot* but if you're porting a project over, you'll have to watch for it!
- Can not be used as a 'client' with your favorite MCU at the moment (unlike the nRF51), since code all runs natively (making this both a pro or a con depending on your needs).
- Examples from the nRF51 and nRF52 are not compatible with each other since they have entirely different programming models.
- Cannot use Bluetooth Classic

Feather nRF52 (NRF52832)



Your browser does not support the video tag. Adafruit Feather nRF52 Bluefruit LE

$24.95
IN STOCK

ADD TO CART

The Feather nRF52 is a new direction compared to our 32u4 or M0 Bluefruit boards. This Feather has only *one* chip on it - and that chip is both the processor you program and also the **Bluetooth Low Energy radio**. What's nice about this is you can do more powerful stuff, and faster too, because you don't have to manage two chips. It's also lower price and lower power since there's only one processor, and easier to put into sleep modes.

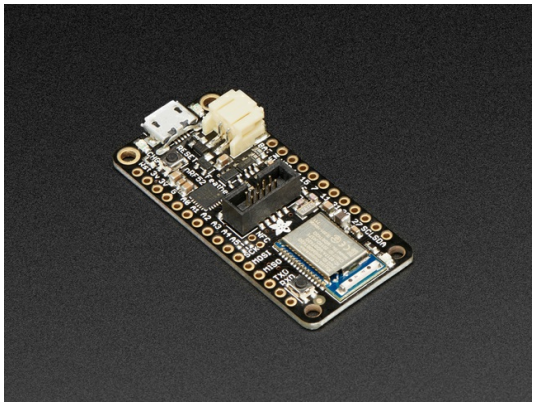
Pros:

- Code runs natively on the nRF52832 MCU, so it's fast, efficient and less expensive than the two MCU based nRF51 boards
- Powerful ARM Cortex M4F processor with 512KB flash and 64KB SRAM
- Better low power potential due to being a single chip solution
- Because everything runs natively, a lot of complexity can be hidden behind simple helper classes, making things like ANCS (https://adafru.it/wfj) possible and manageable compared to the nRF51.
- Supports both Central and Peripheral mode, Central mode meaning it can behave like a phone and initiate connections to other peripherals, although Central SW support is in the very early stages at this point
- Much more room to grow due to the MCU capabilities and flash/SRAM size compared to the earlier nRF51832, and more advanced on board peripherals.

Cons:

- A new chipset that is not an ATmega328 or ATmega32u4, so some older low-level Arduino libraries and techniques are not compatible. This doesn't happen *a lot* but if you're porting a project over, you'll have to watch for it!
- Can not be used as a 'client' with your favorite MCU at the moment (unlike the nRF51), since code all runs natively (making this both a pro or a con depending on your needs).
- Examples from the nRF51 and nRF52 are not compatible with each other since they have entirely different programming models.
- Cannot use Bluetooth Classic
- UART pins (RX/TX) are not available for connecting to sensors/wings, as they are used for programming and debug of the native chip.

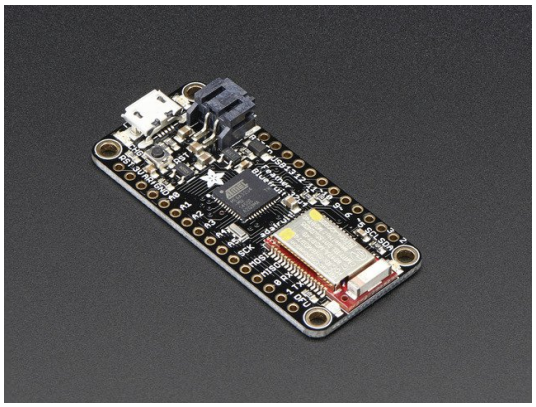
We also carry a special Feather - unlike the rest of the Feather family, this board is not for use with Arduino IDE. Instead, it is for use with Mynewt only! We have programmed it with the Mynewt bootloader and updated the hardware to add an SWD connector and an additional DFU button. **This Feather is for advanced users only, you will be interacting with the Mynewt RTOS rather than Arduino,** and you cannot easily go back-and-forth without an SWD programmer.



Adafruit Feather nRF52 Pro with myNewt Bootloader

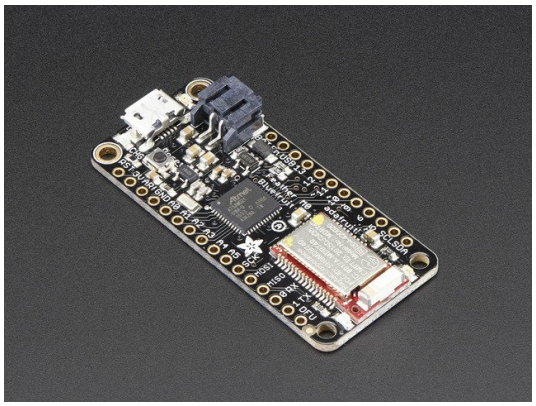
$27.50
IN STOCK
ADD TO CART

Feather Bluefruit M0 and 32u4



Adafruit Feather 32u4 Bluefruit LE

$29.95
IN STOCK
ADD TO CART



Adafruit Feather M0 Bluefruit LE

$29.95
IN STOCK
ADD TO CART

Our original Bluefruit Feathers are the 32u4 and M0-based. These pairs are very similar looking, and have the same basic idea behind them: there is a **main processor** which is an ATmega32u4 or ATSAMD21 and a**co-processor module** which is the red and silver rectangle, containing an nRF51 which can do **Bluetooth Low Energy only**.

The nRF51 is programmed with our Bluefruit firmware, and can be controlled with AT commands over SPI connection. When the main processor (32u4 or M0) wants to send or receive BLE data, it sends commands to the co-processor module. We decided to go this way because the nRF51 did not have as-good a programming setup as the nRF52, and we thought people would have a better experience if the main chip was not also doing the radio work.

That said, if you already have code for the 32u4 or M0 chips, this is an easy upgrade, you get all the peripherals you

know and love. This may be easier for beginners who are not quite ready to work with the native nRF52

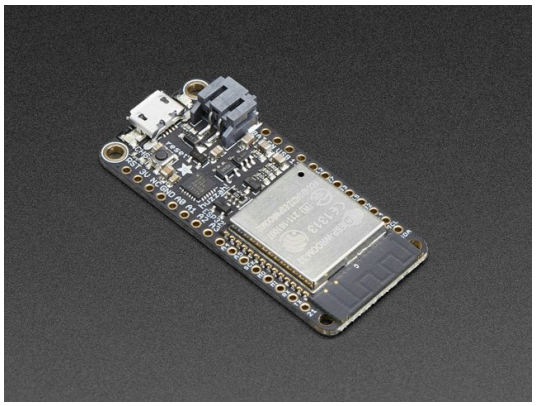
Pros:

- Familiar 32u4 and M0 chipsets
- No need to worry about the Nordic 'softdevice' radio thread interrupting timing-specific code since the radio is completely separate.
- Lots of existing code and projects

Cons:

- Can only act as Peripheral, not Central
- More expensive as it is a 2-chip solution
- Higher power draw
- Can't take advantage of new nRF52 capabilites as the firmware is updated and released by Nordic
- Cannot use Bluetooth Classic

ESP32 Feather



Adafruit HUZZAH32 – ESP32 Feather Board

$19.95
IN STOCK

ADD TO CART

Even though this is primarily considered a WiFi Feather, the ESP32 does contain a BT LE and BT Classic radio! That's right this is the *only* Feather that can do BT classic. It's also the only one that can do WiFi and BT (altho, as of this writing, it cannot do both at once)

Sounds great, right? Well, there's some caveats. As of this writing, May 2017, the ESP32 Bluetooth core is still under development and there's only one basic beacon example for use with the Arduino IDE core. So we put this one at the bottom of the list. You can check the ESP32 Arduino core and what has been added here, (https://adafru.it/wfk) if there's an example for what you want to do, then you're in luck!

Once there's more working examples, we can fill out the pros and cons more. Right now we think it should only be used by bleeding-edge developers who are cool with running weekly regression tests on their project.

Pros:

- All-in-one Feather can do anything and everything (except make you a sandwich)
- Fast dual-core processor
- Low cost

Cons:

- Still waiting on integration and support for BT and BTLE radio

Cellular Feathers

The cool thing about cellular connectivity is you can use it just about anywhere in the world. The FONA GSM module can be used anywhere that 2G GSM exists. In the USA this is provided by T-Mobile so if you have T-Mobile coverage, you have GSM.

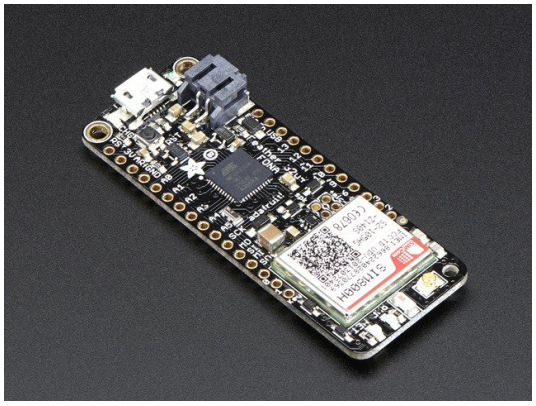
We provide cellular connectivity via the SIM800H module. This module uses AT commands and the main processor, the ATmega32u4, communicates via a UART connection. The AT commands are somewhat standard and they are all documented pretty well, but if you want to do something special that isn't already written up in our library, it can be a little bit of an adventure.

- Compared to WiFi, cellular requires much more power, and has lower data transfer rate. SSL support is not very clearly documented
- Compared to BTLE, data rate is about the same, but power requirement is waaaay higher
- Compared to LoRa or Packet radio, the data rate is about the same, but the power requirements are much higher. That said, you don't need to create your own radio network, you can just use the cellular net that already exists. There's no inherent range limitation, as long as GSM coverage is present.

Often times, the only real alternatives to cellular are WiFi or LoRa. WiFi is a good alternative - but only if you can be sure that there's a WiFi base-station or hotspot nearby. LoRa can do long range communication, up to a few KM but you need to set up your own network/gateway.

Cellular Feathers

This section is fairly short because there's only one cellular-capable Feather at this time, the 32u4 FONA



Adafruit Feather 32u4 FONA

$44.95
IN STOCK

ADD TO CART

Pros:

- Can use (just about) anywhere in the world!
- Surprisingly low cost of entry
- Gain access to SMS network, phone calls, and Internet through GPRS
- Basic geo-location capability built-in via cell-tower triangulation (not as good as GPS but good to within ~1 mile)
- No need for a base-station or hotspot
- ATmega32u4 is well supported chip, with native USB and full Arduino core.

Cons:

- You'll pay per message, call, text or megabyte. A SIM card and plan is required

- GSM is only supported by T-Mobile in the USA, and GSM will be supported until at least 2020 but after that, you'll want to upgrade to LTE
- High power requirements, a large battery is required to handle the 2 Amp power spikes when transmitting/connecting to the network
- Cellular connectivity can be spotty, especially when roaming. Code needs to be able to manage re-connection.
- AT commands can be a little clunky, require care and parsing.
- Extra large Feather to make space for the module
- Antenna placement often trips people up: if it's too close to the main processor it can reset the Feather, so you can't make it *too* compact.
- Cellular module runs directly off of LiPo battery so you cannot use only USB power

LoRa & Radio Feathers

WiFi, Bluetooth, ZigBee and Cellular are everyday wireless protocols you use every day - at home, at work, with friends. Sometimes, though, you need to leave the well-known and venture to the more exotic! The RFM69 and LoRa Feathers let you create a more flexible wireless network.

Instead of depending on a WiFi router, Bluetooth Central, ZigBee Master or Cellphone tower, LoRa and RFM69 work well in point-to-point or meshing configurations, with eye-popping ranges. They also don't need complex pairing or connection overhead, turn on the radio and transmit when you like, then turn it back off. They are the "rugged individuals" of wireless!

The trade-off is that you'll have to do a bit more work in the software to manage your packets, and re-transmission.

Pros:

- No need for master or central router, you can use two radios or hundreds of radios, all talking to each other
- Long range: RFM69 can go 500 meters with a basic antenna, 5 km with directional antennas and care. LoRa can go 2 km with a basic antenna, 20 km with directional antennas. (Contrast with ZigBee & WiFi's ~100m max range, and BT ~10m range)
- Great for short burst packets, with ~10Kbps rate
- Best for outdoor usage
- RFM69 have low cost, LoRa are more expensive (but with longer range)
- Adjustable transmission power
- No link overhead, send packets without needing setup/teardown of connection
- RFM69 has built-in encryption capability

Cons:

- Neither RFM69 nor LoRa are standardized, or built into laptops, so you need at least two of the radios to connect to a computer
- LoRa Feathers do not have LoRaWAN built in, must be added in software (which isn't hard)
- Link management and re-transmission is up to the user, so there's a little more work to do!
- LoRa radio does not have built-in encryption, must be added by hand by the user.
- Not good for audio/photo or video bandwidths, best for small data packets.
- Tweaking and tuning of antenna setup required to get the longest ranges.

Which One???

There's a lot of options for these radio feathers, because each chipset (32u4 and M0) has two radio types (RFM69 and LoRa) and then two frequencies (433MHz and 900MHz). That's 8 total options.

The easiest decision is what frequency you need to use. If you are in ITU "Americas" you should go with the 900MHz radio. If you are in ITU "Europe" you should go with 433MHz radio. (If you are a HAM radio licensed operator you may be able to use either, check your countries' available frequency bands, and restrictions)

If you are in a country that allows unlicensed usage of 868 MHz, pick the 900 MHz radio. Both RFM69 and LoRa are 'software tuned' - so just make sure you set the Arduino library code to 868 MHz when you upload it.

You can tell what frequency your radio is by the dot on top. Red dot means ~400 MHz, Green dot means ~900 MHz.

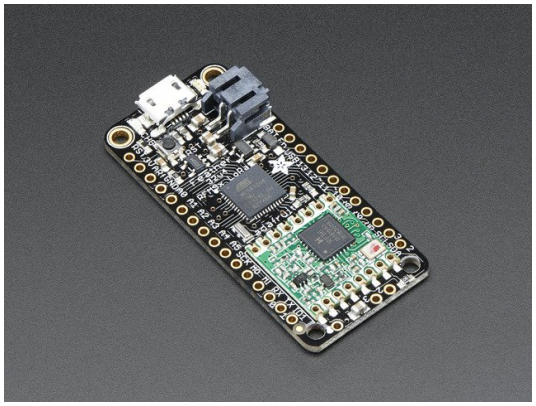
Then pick your radio type. The lower cost RFM69 doesn't go as far but has a lot of functionality and can do the job for

most projects. LoRa has much better range, and can be used with LoRaWAN, but has additional cost. You cannot mix and match radios: RFM69 cannot 'speak' LoRa.

Both radio modules are identical shape and pinout, so we re-use the same Feather layout for both.

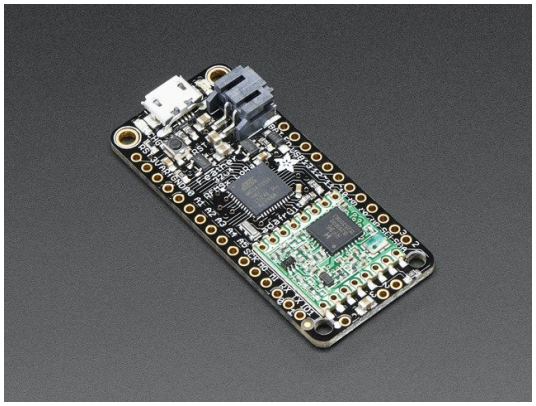
Finally, pick your chipset. ATmega32u4 is a well known 8-bit AVR processor, and may be more familiar to people who use the similar Arduino '328s. The ATSAMD21 M0 is a 32-bit Cortex M0, with a ton more flash and RAM, and is also supported in Arduino but is a little newer and so does not have as many years of projects and example code already. If you want to use LoRaWAN firmware, we suggest M0 Feathers since you will need to store all that extra software

LoRa Radio Feathers



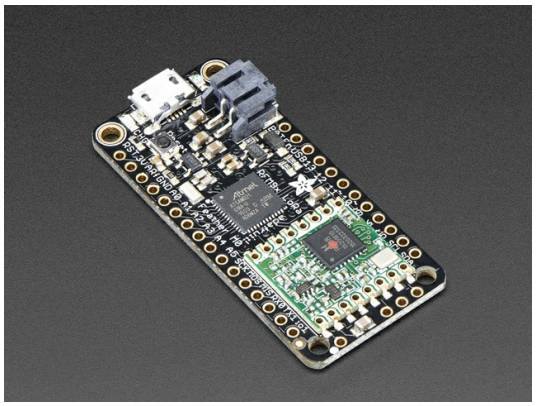
Adafruit Feather 32u4 RFM96 LoRa Radio - 433MHz

$34.95
IN STOCK
ADD TO CART



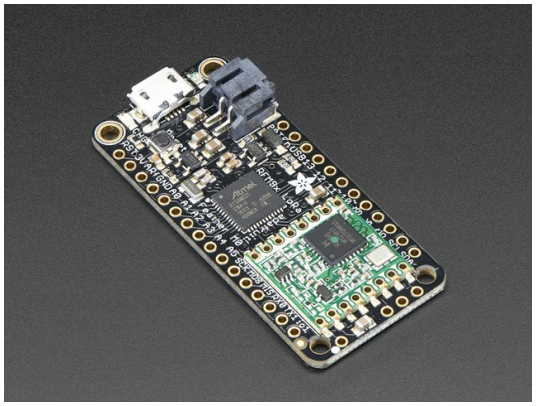
Adafruit Feather 32u4 RFM95 LoRa Radio- 868 or 915 MHz

$34.95
IN STOCK
ADD TO CART



Adafruit Feather M0 RFM96 LoRa Radio - 433MHz

$34.95
IN STOCK
ADD TO CART

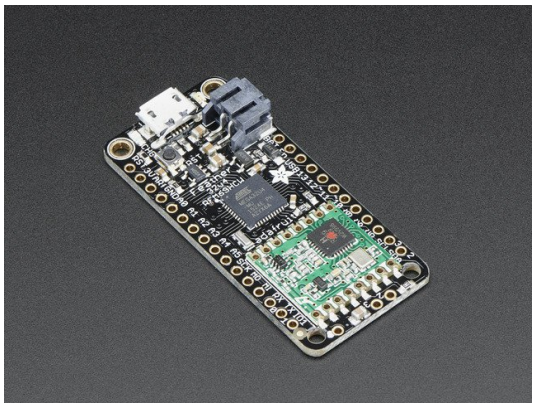


Adafruit Feather M0 with RFM95 LoRa Radio - 900MHz

$34.95
OUT OF STOCK

OUT OF STOCK

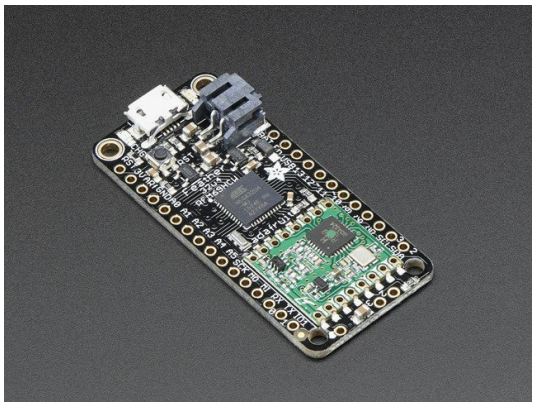
RFM69 Radio Feathers



Adafruit Feather 32u4 with RFM69HCW Packet Radio - 433MHz

$24.95
IN STOCK

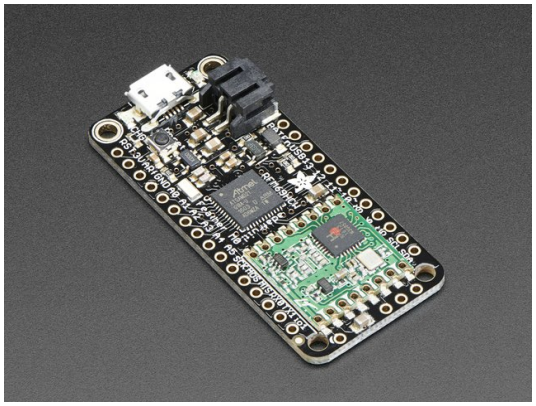
ADD TO CART



Adafruit Feather 32u4 RFM69HCW Packet Radio - 868 or 915 MHz

$24.95
IN STOCK

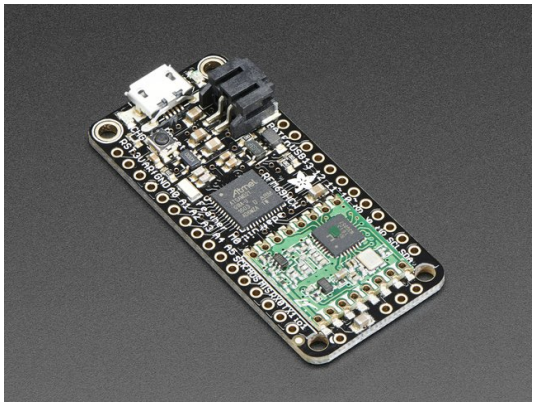
ADD TO CART



Adafruit Feather M0 RFM69HCW Packet Radio - 433MHz

$24.95
IN STOCK

ADD TO CART

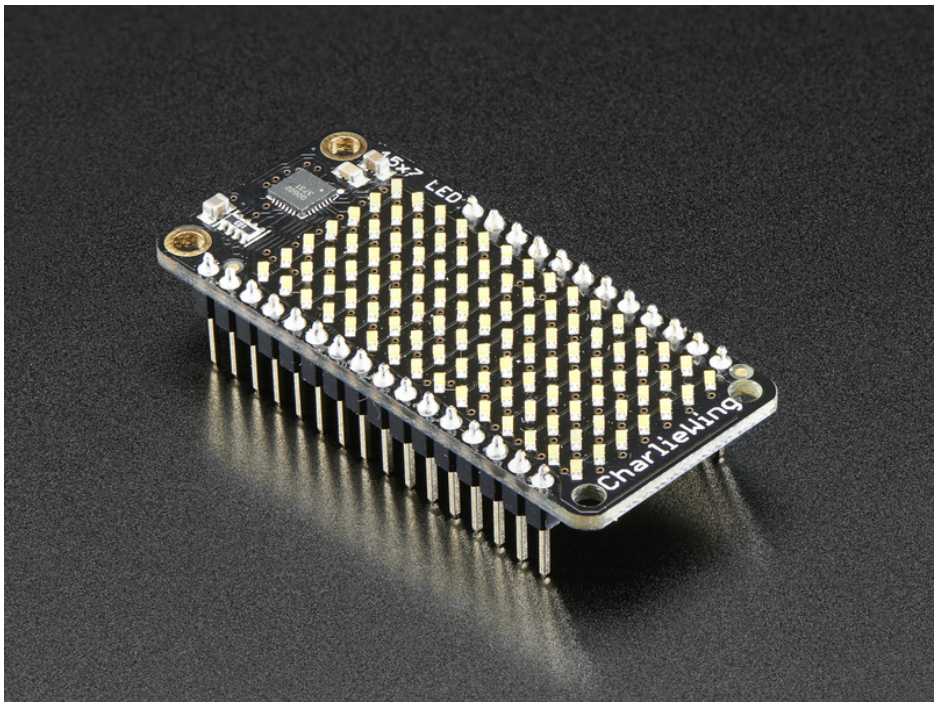


Adafruit Feather M0 RFM69HCW Packet Radio - 868 or 915 MHz

$24.95
IN STOCK

ADD TO CART

FeatherWings



Giving your Feather a Wing lets it fly!

The best part about the Feather system is all the wings you can add on.

Start with a WiFi feather and add an TFT for a WiFiWeather display (https://adafru.it/wfm).

Or, take a LoRa feather and add an OLED display to make aRemote Control Effects Box for theatrical control (https://adafru.it/wfn)

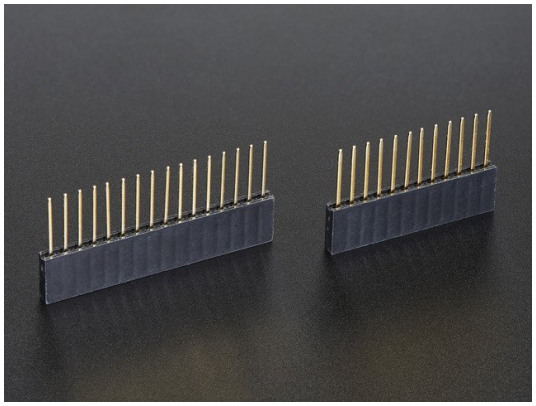
Pick and choose - you can even add multiple wings (as long as there are no pin or I2C address conflicts). We tried to use I2C for as many Wings as possible, so you can make fairly advanced projects just by plugging boards together.

Accessories

Here are some accessories you may find handy as you add Wings to your Feather

Check the list on the left navbar, or just keep going through the guide, to see the wide variety of Wings we have for you

Stacking Headers

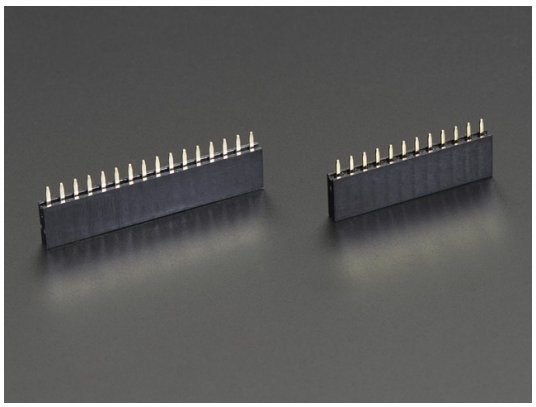


Stacking Headers for Feather - 12-pin and 16-pin female headers

$1.25
IN STOCK
ADD TO CART

You can either stack Feathers by soldering stackable headers onto the Feather and Wings, or you can have them side by side in a Multiplier Wing. Check the Proto/Multiplier wing page for more details (https://adafru.it/wfo)

Female Headers

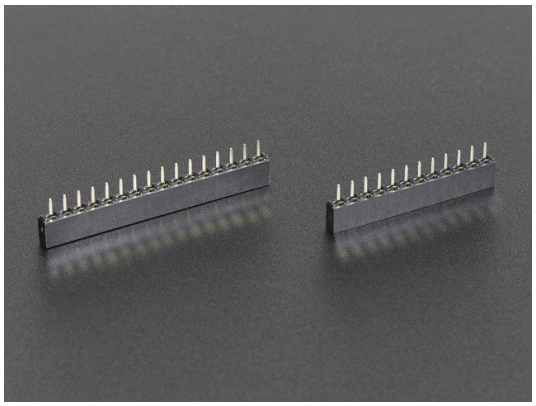


Header Kit for Feather - 12-pin and 16-pin Female Header Set

$0.95
IN STOCK
ADD TO CART

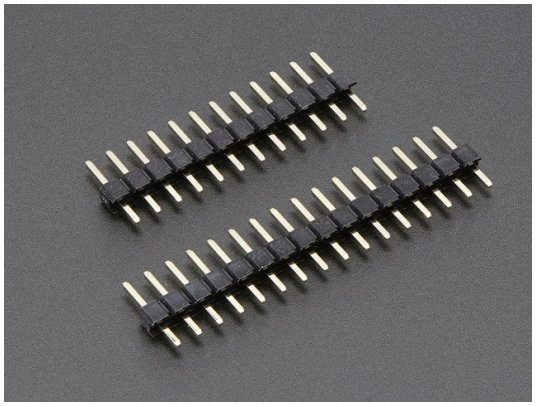
If you don't need to use your Feather in a breadboard, you can use these plain female headers along with plain male headers, to create a cute sandwich. Great for wearable or enclosed projects!

Shorty Headers



Short Headers Kit for Feather - 12-pin + 16-pin Female Headers

$1.50
IN STOCK
ADD TO CART

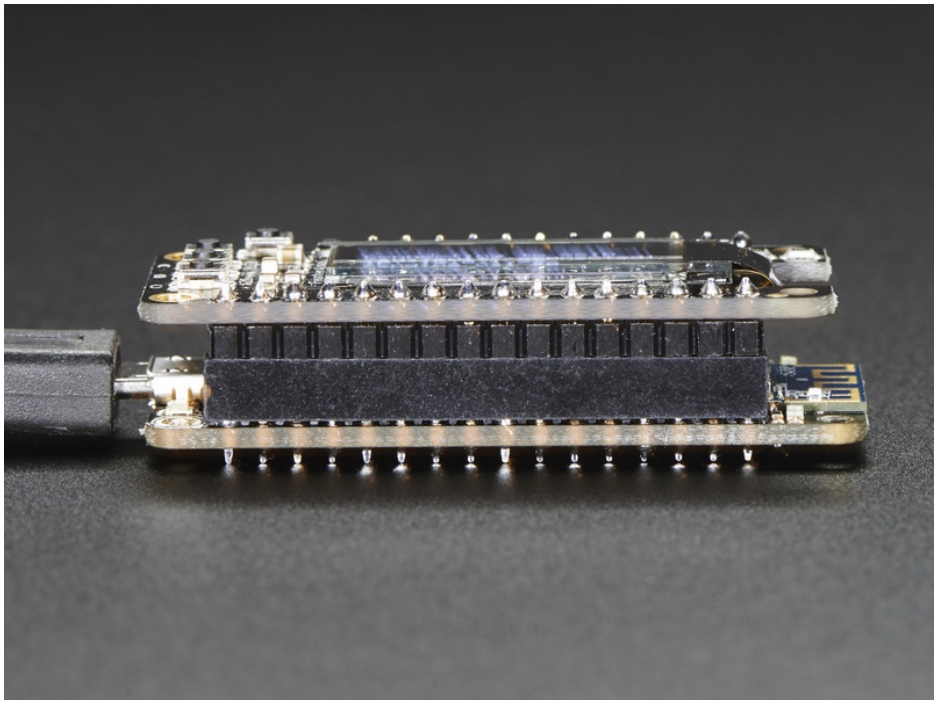


Short Feather Male Headers - 12-pin and 16-pin Male Header Set

$0.50
IN STOCK

ADD TO CART

If you want to make a very compact package of two Feathers, you can use short headers. These are good for when you need the very slimmest setup.



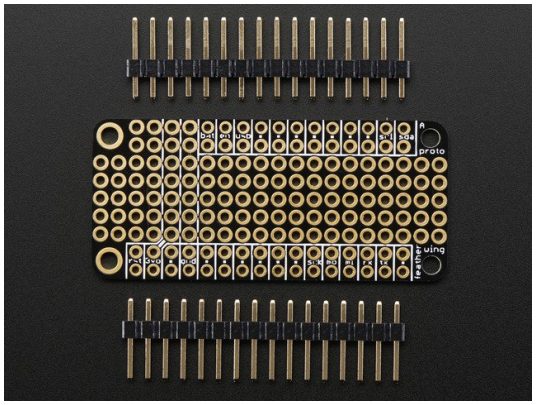
Just remember to pick these up before you start soldering stacking or 'standard' headers on!

Proto & Multiplier Wings

Our most popular Wings are the simplest. These are perfect when you want to do something custom, taking it beyond just a Feather+Wing combo!

All of these Wings are compatible with all Feathers. Because they don't contain any active circuitry, they can also be used with any Wings

Proto Wing



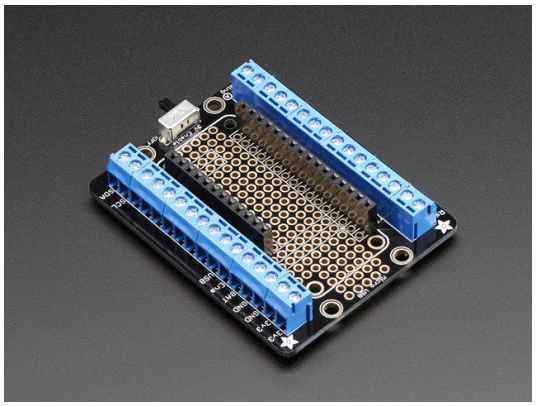
FeatherWing Proto - Prototyping Add-on For All Feather Boards

$4.95
IN STOCK

ADD TO CART

Nice and simple, you get a 0.1" prototyping area and a breakout for each pin of your Feather. Add any chips or sensors you like! There's a bunch extra 3V and GND pins in a row as well.

Terminal Block Proto Wing



Assembled Terminal Block Breakout FeatherWing for all Feathers

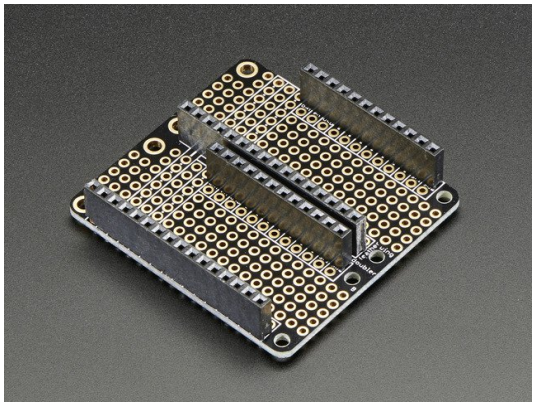
$14.95
IN STOCK

ADD TO CART

This is like a super hard-core version of the proto wing. Not only do you get the same exact prototyping area as the proto wing, but you *also* get terminal block breakouts for all of the pins, *and* there's an on-off switch that will turn off the Feather with a flick.

And, it comes fully assembled! No soldering is required, so you can wire up your project super fast.

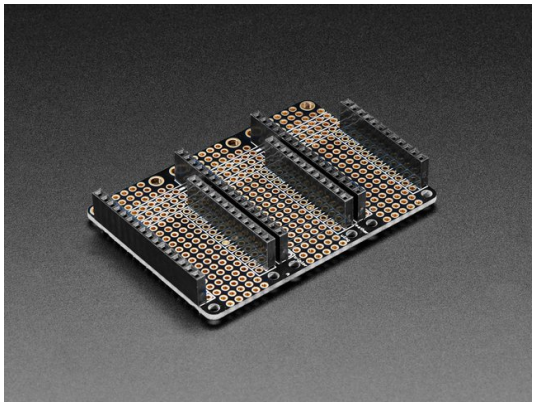
Doubler and Tripler



FeatherWing Doubler - Prototyping Add-on For All Feather Boards

$7.50
IN STOCK

ADD TO CART



FeatherWing Tripler Mini Kit - Prototyping Add-on For Feathers

$8.50
IN STOCK

ADD TO CART

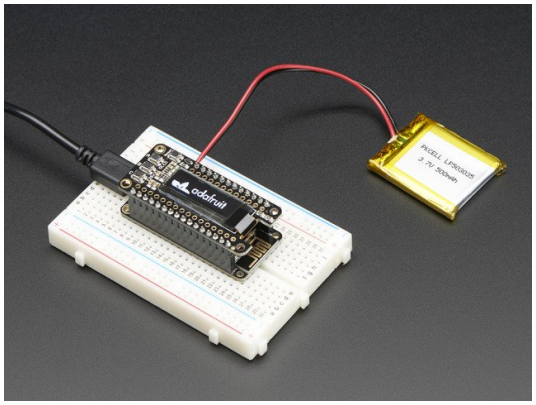
If you don't want to stack 'up' you can stack side-by-side. These two wings give you all that prototyping area you could possibly need, and 'duplicate' pinouts for two or three Feather boards. With the doubler you can have a Feather and Wing. With the tripler, you can have a Feather and two wings. Of course, you can also use stacking headers to grow up too, for lots of possibilities.

Graphic Display Wings

Display Wings let you add a monochrome or color display for lots of data. These are very popular, as you can imagine, for adding a user interface.

These Wings work with all Feathers.

You can only really use one display at a time. The OLED Featherwing uses only I2C and works with any other Wings. The TFT 'wing uses a lot of pins, so works best with only I2C Wings.



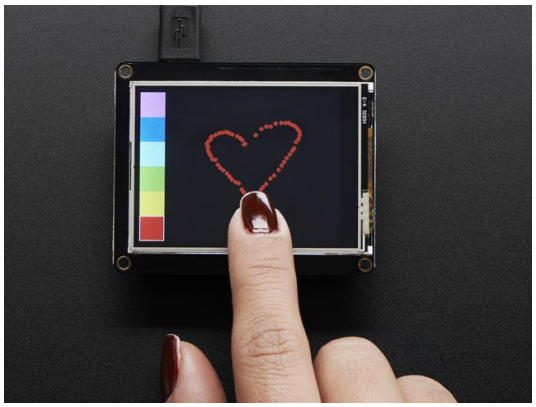
Adafruit FeatherWing OLED - 128x32 OLED Add-on For Feather

$14.95
IN STOCK
ADD TO CART

This tiny Wing is so cute, and it has a 128x32 monochrome OLED display. OLEDs have a crisp and high-contrast look, so they are great for indoor and some outdoor use. There's no backlight so they use little power (only as much as the pixels that are lit).

The OLED is connected over I2C so it uses only two pins. In addition, there are 3 little buttons connected directly to three GPIO pins. You can use the buttons to add basic interactivity. Only one OLED FeatherWing can be used with a Feather board.

This feather has a 0.96" 160x80 Color TFT Display with 16-bit full color capability. And, so you can make a proper UI, it has a 5-way navigation switch and two push buttons. The joystick can go left, right, up, down and 'in' for selection. Two buttons on the side can change modes or whatever you like.



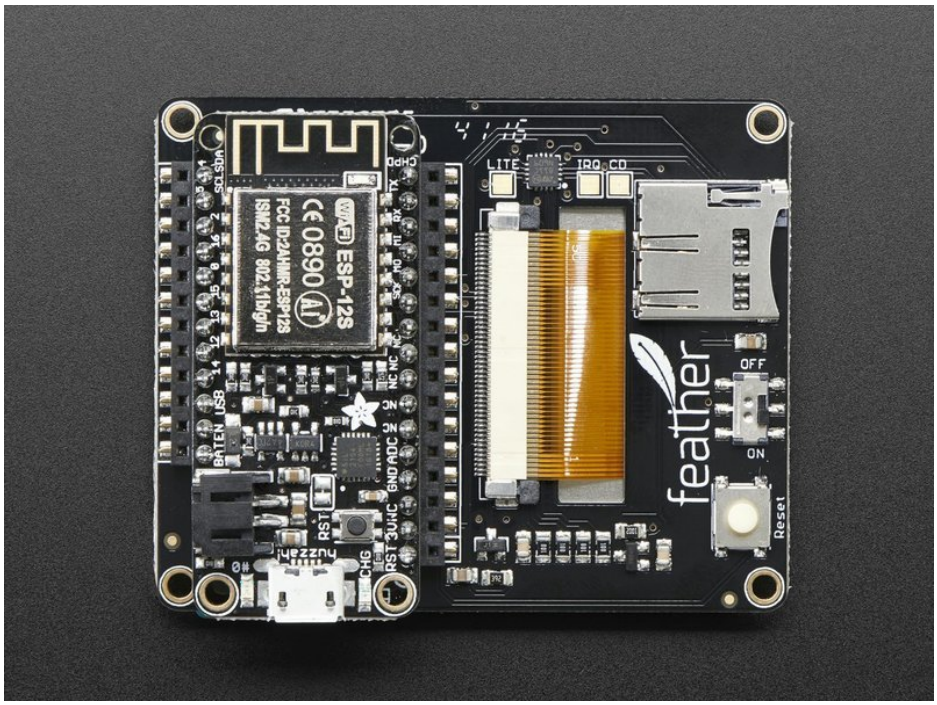
TFT FeatherWing - 2.4" 320x240 Touchscreen For All Feathers

$29.95
IN STOCK
ADD TO CART

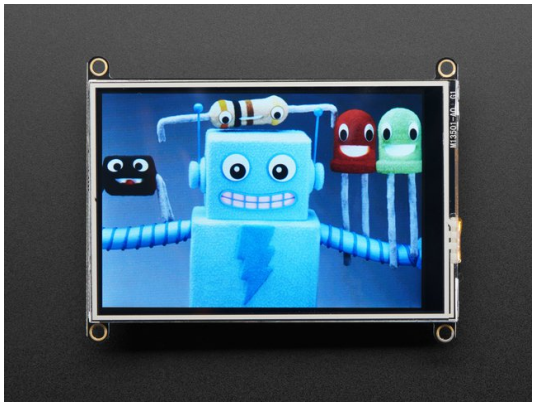
If you need a bigger and more colorful screen, the TFT FeatherWing has just about everything. There is a 320x240 full-color TFT display. There is also a resistive touchscreen on top, for interactivity. On the bottom is a microSD card holder so you can store images or data to be displayed. We also toss in an on-off switch.

The display is much bigger than the OLED, and uses the SPI port for communication with the TFT/touch/SD card.

The Wing requires an additional two pins for the TFT. One additional pin if you are using the touch screen, and one extra pin if using SD interface. (So, SPI + 4 GPIO pins max if using all three)



Unlike other wings, this one is fully assembled and the feather plugs into the bottom. If you want to stack Wings, you'll need to solder stacking headers to the Feather, then stack on top of that.



Adafruit TFT FeatherWing - 3.5" 480x320 Touchscreen for Feathers

$39.95
IN STOCK

ADD TO CART

Spice up your Feather project with a beautiful 3.5" touchscreen display shield with built in microSD card socket. This TFT display is 3.5" diagonal with a bright 6 white-LED backlight. You get a massive 480x320 pixels with individual 16-bit color pixel control. It has way more resolution than a black and white 128x64 display, and twice as much as our 2.4" TFT FeatherWing. As a bonus, this display comes with a resistive touchscreen attached to it already, so you can detect finger presses anywhere on the screen

This FeatherWing uses a SPI display, touchscreen and SD card socket. **It works with any and all Feathers** but given the large display it works *best* with our faster boards like the nRF52, ESP8266, ESP32, M0, M4, WICED, and Teensy. We also include an SPI resistive touchscreen controller so you only need one additional pin to add a high quality

touchscreen controller. One more pin is used for an optional SD card that can be used for storing images for display.

This Wing comes fully assembled with dual sockets for your Feather to plug into. You get two sockets per pin so you can plug in wires if you want to connect to Feather pins. Alternatively, each pin has a large square pad on the PCB for direct soldering.

LED Display Wings

If you don't need a full graphical display, you can use our LED matrix/segment Wings. These can be good for simplified displays where you don't need the complexity and pins of a graphical TFT or OLED. They all use I2C only, so they won't take up extra pins or interfere with other Wings if you want to stack them up or use them side-by-side.

We have a **ton** of options, with various displays and colors!

These Wings use I2C only for the interface, and work with all Feathers. You can change the addresses used to have multiple displays, and they can be used with other Feathers as well - just make sure there are no I2C address conflicts.

Seven Segment Wing

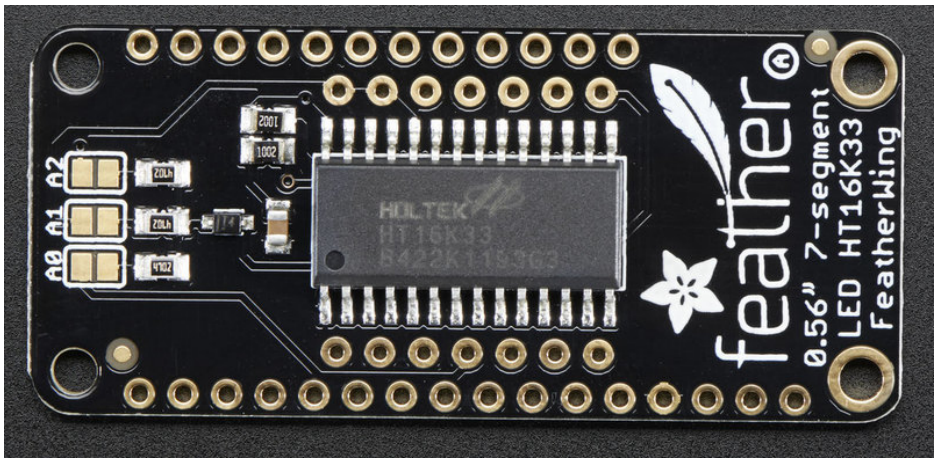
Your browser does not support the video tag.

Adafruit 0.56" 4-Digit 7-Segment FeatherWing Display

$9.95
OUT OF STOCK

OUT OF STOCK

Just want to display some numbers? The 7-Segment FeatherWing has 4 digits and looks nice and bright. Comes in 5 bright colors!



Each Wing has an HT16K33 chip, which does all the multiplexing for you. The address is 0x70 but can be changed to 0x70-0x77 so up to 8 displays can be used at once.

Alphanumeric (14-segment) Wing

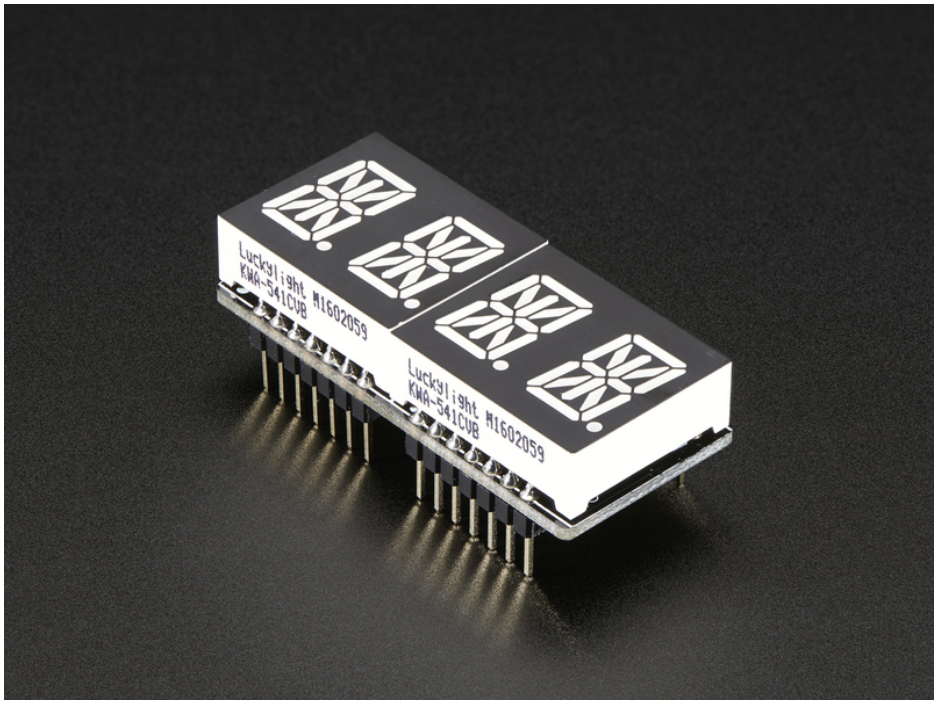
Your browser does not support the video tag.

Adafruit 0.54" Quad Alphanumeric FeatherWing Display - Various

$0.00
OUT OF STOCK

OUT OF STOCK

This Wing is a step up from just 7-segments. With 2x as many segments! You can have both letters and numbers displayed, with big bright digits. Comes in 6 different color varieties



Like the 7-segment, each Wing has an HT16K33 chip, which does all the multiplexing for you. The address is 0x70 but can be changed to 0x70-0x77 so up to 8 displays can be used at once.

8x16 Matrix FeatherWing

Your browser does not support the video tag.

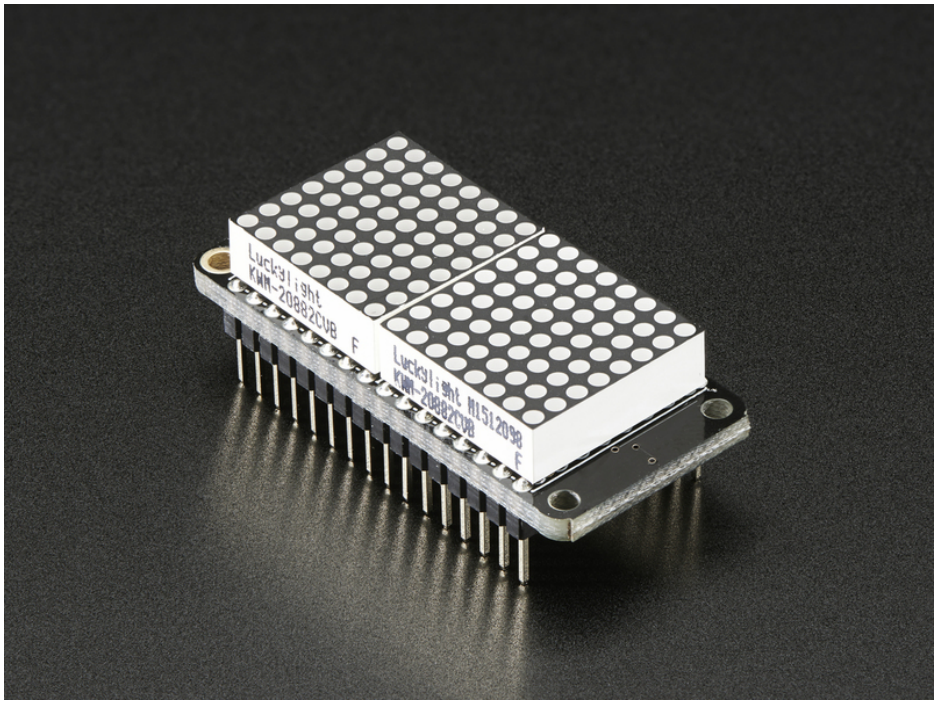
Adafruit 0.8" 8x16 Matrix FeatherWing Display Kit Various Colors

$11.95
OUT OF STOCK

OUT OF STOCK

The most segments possible! This Wing has two tiny 8x8 matrices side-by-side to make an 8x16 grid. Great for little scrolling displays. These have a nice diffusion for a lovely dotted look.

Comes in 6 colors.



Like the 7-segment and alphanumeric, each Wing has an HT16K33 chip, which does all the multiplexing for you. The address is 0x70 but can be changed to 0x70-0x77 so up to 8 displays can be used at once.

Charlieplexed LED Matrices

Your browser does not support the video tag.

Adafruit 15x7 CharliePlex LED Matrix Display FeatherWings

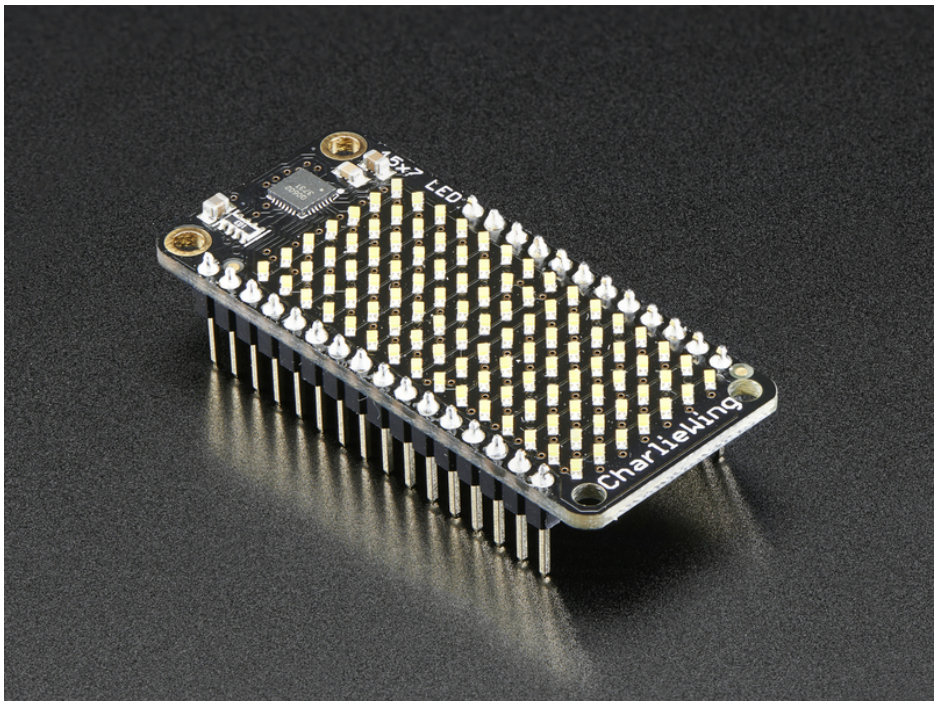
$0.00
OUT OF STOCK

OUT OF STOCK

The Charlieplexed wings are *similar* to the 8x16 wings above, but use a different chipset. Instead of the HT16K33, we use the IS31FL3731. Now, you might be wondering, who cares what chipset is used? But there's some things you can do with the IS31FL3731 that you can't with the HT16K33.

Unlike the simple HT16K33, the IS31 can drive the individual LEDs in a grid at different brightnesses. This means you get a full 8-bit PWM individual brightness per pixel. This is great for when you want to make artistic LED displays that

aren't just 'on and off'. But, the IS31 requires the LEDs to be in a certain arrangement, so we use individual 0603 LEDs rather than the 8x8 modules above



The tradeoff is you get individual LED control, and a sparkly look, but not the nice diffusion of the 8x16. Also, there's smaller number of LEDs because we couldn't fit a full 8x16 grid.These come in 6 colors (warm white isn't shown above). The I2C address can be set to 0x74 or 0x77 so you can, in theory, have two of these at a time.

RGB Pixel Wings

The NeoPixel FeatherWing requires only one pin, and works with any Feather - although you may need to change the pin used by soldering the bottom jumper. You can use as many as you like, by assigning a different pin to each FeatherWing - just watch out for your power usage!

Your browser does not support the video tag.

NeoPixel FeatherWing - 4x8 RGB LED Add-on For All Feather Boards

$14.95
IN STOCK
ADD TO CART

Pros:

- Blindingly bright RGB pixels! So much color!
- Familiar NeoPixels are well known, lots of example code available for all chips
- Uses only one pin for all of the LEDs

Cons:

- Somewhat low resolution display, 4x8 pixels
- Can be power hungry, up go 35mA per pixel at full white - so proper programming is essential
- LEDs are powered off of USB/LiPo, so disabling the Feather via the EN pin will not disable the NeoPixels.

Dotstar LEDs need two data pins, but they can be clocked faster than NeoPixels.

Your browser does not support the video tag.

Adafruit DotStar FeatherWing - 6 x 12 RGB LEDs

$29.95
IN STOCK
ADD TO CART

This is the **DotStar FeatherWing, a 6x12 RGB LED Add-on For All Feather Boards**! Using our Feather Stacking Headers (http://adafru.it/2830) or Feather Female Headers (http://adafru.it/2886) you can connect a FeatherWing on top or bottom of your Feather board and make your Feather board strut like a peacock at a rave.

Put on your sunglasses before staring into these 72 configurable RGB LEDs, they are super bright! Arranged in a 6x12

matrix, each 2mm by 2mm sized RGB pixel is individually addressable. Only two pins are required to control all the LEDs. On the bottom we have jumpers for the Data and Clock lines so you can change them from the defaults. **Works with any/all of our Feathers!** You can cut the default jumper traces and use any pins you like

Your browser does not support the video tag.

Adafruit RGB Matrix Featherwing Kit - For M0 and M4 Feathers

$7.50
IN STOCK
ADD TO CART

Create a dazzling light up project with our new **RGB Matrix FeatherWing**. Now you can quickly and easily create projects featuring your favorite 16 or 32-pixel tall matrix boards. Using our RGB Matrix library is easy and works wonderfully with any of our M0 or M4 based Feathers.

Please note: This wing is only tested/designed to work with the SAMD21 M0 and SAMD51 M4 Feathers. It's not for use with any other Feathers at this time. (That said, if you'd like to add support, we'd be happy to take a pull request (https://adafru.it/aHj) on the library repo)

This wing can be assembled in one of two ways. You can either solder in a 2x8 IDC shrouded header on the top, then plug in the IDC cable that came with your matrix. This makes it easy to stack on top of your Feather. Or, you can solder in the 2x10 socket header on the *bottom* of the Wing, and then stack your Feather on top. That way you can *plug it directly into the back of the matrix* *mind blown*

This FeatherWing will work great with any of our 16x32 (https://adafru.it/CIE), 32x32 (https://adafru.it/CIF) or 64x32 (https://adafru.it/CIG) RGB matrices, and is definitely the easiest way to glow and go.

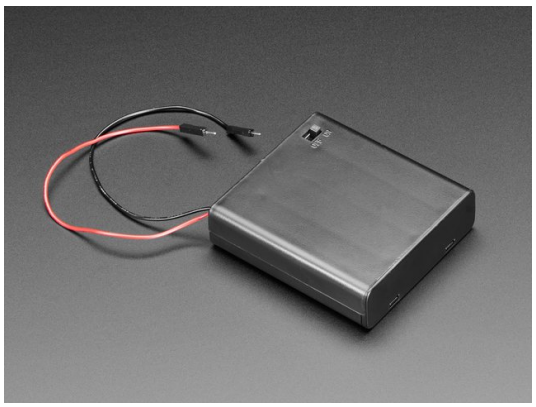
Motor & Servo Wings

Feather can make a great base for your next robotic project. We've got three Wings that make easy work of adding motors.

Best of all, **these FeatherWings only use I2C, so they work with all Feathers**. The I2C address can be changed with a solder jumper, so you can add multiple Wings for lots of motors, or use them with any other Wings. (Just make sure you have each wing with a unique I2C address)

Both of these Wings are designed for driving fairly powerful motors, easily 1A draw each. For that reason, we do not run the motors off of USB or the LiPo battery - they are dedicated for running the Feather only. Instead, the 'loud and messy' power supply for the motors is provided via a terminal block.

We recommend something like our 4 x AA battery pack, with NiMH rechargeable batteries.

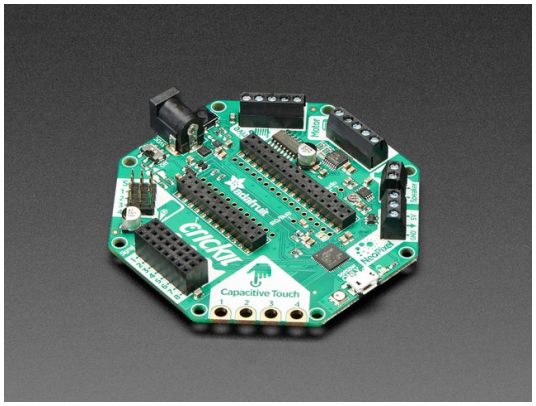


4 x AA Battery Holder with On/Off Switch

$2.95
IN STOCK

ADD TO CART

Adafruit CRICKIT Controller



Adafruit CRICKIT FeatherWing for any Feather

$29.95
IN STOCK

ADD TO CART

Crickit is our **C**reative **R**obotics & **I**nteractive **C**onstruction **Kit**. It's an add-on to the popular Feather ecosystem that lets you **#MakeRobotFriend** using CircuitPython, MakeCode, or Arduino.

Plug in *any* Feather mainboard you want into the center, and you're good to go! **The Crickit is powered by seesaw, our I2C-to-whatever bridge firmware. So you only need to use two I2C data pins to control the huge number of inputs and outputs on the Crickit. All those timers, PWMs, sensors are offloaded to the co-processor.**

The only thing that is *not* managed by seesaw is the audio output. We provide a small jumper you can solder to connect the audio amplifier to the first analog pin. On our Feather M0's this is a true analog output (DAC) and you can play audio clips with CircuitPython or Arduino. Other Feathers *may not have a DAC!* In that case, you can solder a wire to jumper the audio amp to a PWM pin.

You get to use all the non-I2C signal pins on your feather and get a boat-load of extra in/out pins, motor controllers, capacitive touch sensors, a NeoPixel driver and amplified speaker output. It complements & extends your Feather so you can still use all the goodies, including stacking FeatherWings on top. But now you have a robotics playground as well.

You get:

- 4 x Analog or Digital Servo control, with precision 16-bit timers
- 2 x Bi-directional brushed DC motor control, 1 Amp current limited each, with 8-bit PWM speed control (or one stepper)
- 4 x High current "Darlington" 500mA drive outputs with kick-back diode protection. For solenoids, relays, large LEDs, or one uni-polar stepper
- 4 x Capacitive touch sensors with alligator-pads
- 8 x Signal pins, digital in/out or analog inputs
- 1 x NeoPixel driver with 5V level shifter - The NeoPixels are buffered and controlled by the seesaw chip
- 1 x Class D, 4-8 ohm speaker, 3W-max audio amplifier - the audio input pin is available as a solder-able pad for your configuration, you can connect it to your Feather's DAC or PWM output as you desire.

Servo & PWM Control

Your browser does not support the video tag.

8-Channel PWM or Servo FeatherWing Add-on For All Feather Boards

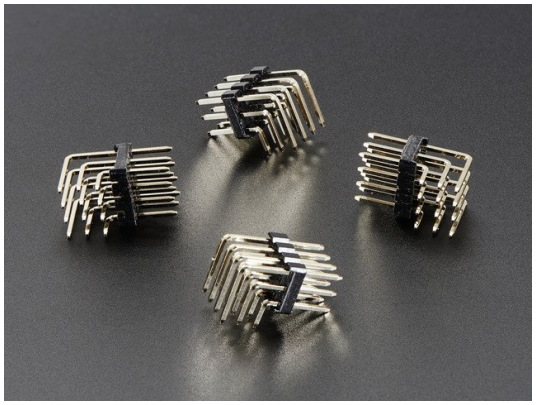
$9.95
IN STOCK
ADD TO CART

Most of our Feathers do have PWM/Servo outputs, but often they are shared or there's not enough of them. With this Wing you need not worry about jitter or timer allocation. The 8 x servo outputs are completely driven by a I2C-controlled chip. All the outputs have the same PWM rate, but have individually controlled duty cycles @ 12-bit resolution.

- There's an I2C-controlled PWM driver with a built in clock. That means that, unlike the TLC5940 family, you do not need to continuously send it signal tying up your microcontroller, its completely free running!
- It is 5V compliant, which means you can control it from a 3.3V Feather and still safely drive up to 6V outputs (this is good for when you want to control white or blue LEDs with 3.4+ forward voltages)
- 6 address select pins so you can stack up to 62 of these on a single i2c bus, a total of 992 outputs - that's a lot of servos or LEDs
- Adjustable frequency PWM up to about 1.6 KHz

- 12-bit resolution for each output - for servos, that means about 4us resolution at 60Hz update rate
- Configurable push-pull or open-drain output

If stacking these, we recommend using right-angle servo connectors:



3x4 Right Angle Male Header - 4 pack

$2.95
OUT OF STOCK

OUT OF STOCK

Brushed Motor and Stepper Controller

Your browser does not support the video tag.

DC Motor + Stepper FeatherWing Add-on For All Feather Boards

$19.95
IN STOCK

ADD TO CART

Motors require both GPIO and PWMs, and they quickly start hogging all your pins up. With this Wing, you get 4 full H-bridges, so you can control up to 4 DC motors, 2 steppers, or 1 stepper and 2 motors.

Motor FeatherWing Specs:

- 4 full H-Bridges: the TB6612 chipset provides **1.2A per bridge** with thermal shutdown protection, internal kickback protection diodes. Can run motors on 4.5VDC to 13.5VDC.
- **Up to 4 bi-directional DC** motors with individual 12-bit speed selection (so, about 0.02% resolution)
- **Up to 2 stepper motors** (unipolar or bipolar) with single coil, double coil, interleaved or micro-stepping.
- Motors automatically disabled on power-up
- Big 3.5mm terminal block connectors to easily hook up wires (18-26AWG) and power
- Polarity protected 2-pin terminal block and jumper to connect external power, for separate logic/motor supplies
- Completely stackable design: 5 address-select jumper pads means up to 32 stackable wings: that's 64 steppers or 128 DC motors! What on earth could you do with that many steppers? I have no idea but if you come up with something send us a photo because that would be a pretty glorious project.

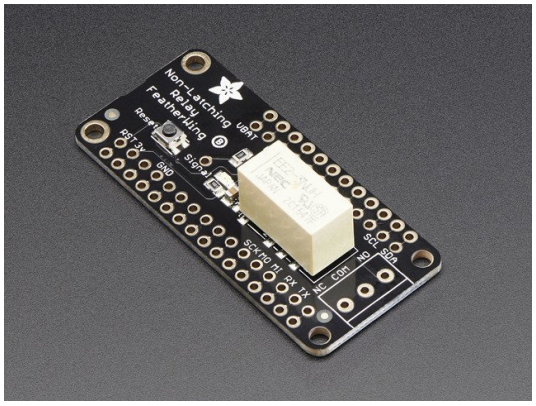
Relay Wings

Click-clack! If you want to control devices or appliances with your Feather, you can do so with a Relay FeatherWing. These Wings have a single relay on them. We have a few different types, but all have similar functionality.

The relay is controlled by one or two pins. You'll need to solder a jumper wire or close a solder jumper to select the pin. Since they're simple, **you can use them with any Feather and along-side any Wing**, as long as the pins you select for relay-control are not already use.

You can also have multiple relays if you like, just make sure the pins selected are unique.

Non-Latching Mini Relay



Adafruit Non-Latching Mini Relay FeatherWing

$7.95
IN STOCK

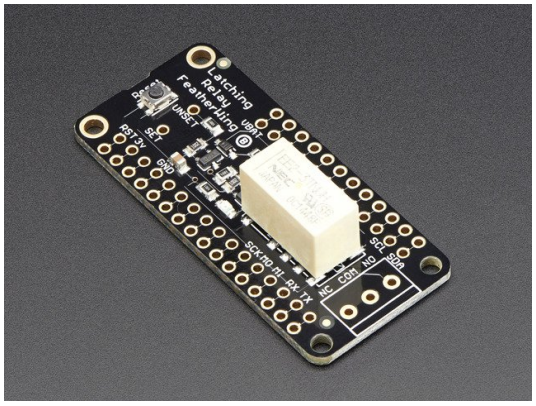
ADD TO CART

This is the simplest relay. Drive the pin high to set the relay, and drive the pin low to unset it.

You can switch up to 2A of resistive current at 30VDC or ~40VAC or lower. At 110VDC you can switch up to 0.3A, at 120VAC up to 0.5A, and at 250VAC you can switch up to 0.6A. Check the datasheet for the relay for the exact switching capacity, and of course, for reactive/inductive loads you will need to de-rate. This isn't a relay you can use to turn on and off your washer/dryer, stick to 60W or less.

50mA current is drawn when the relay is set. If power is lost, the relay will go back to being unset

Latching Mini Relay



Adafruit Latching Mini Relay FeatherWing

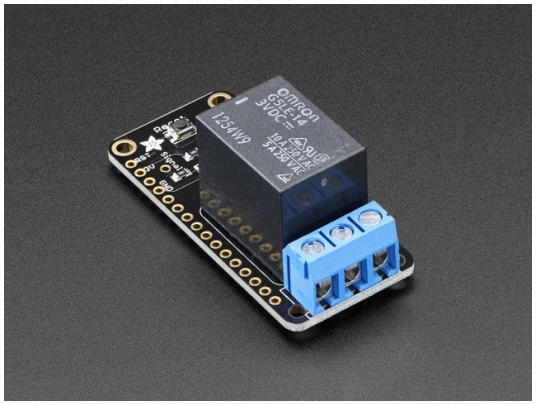
$7.95
IN STOCK

ADD TO CART

This is the **Latching** version of the above relay. This one requires two pins, a **SET** and **UNSET** and instead of keeping the SET pin high, you only have to pulse each pin high for 10ms to latch the relay open or closed. You need two pins but save power. Note, if power is lost, the relay will stay in the last setting.

Both FeatherWings use the same family of relay. You can switch up to 2A of resistive current at 30VDC or ~40VAC or lower. At 110VDC you can switch up to 0.3A, at 120VAC up to 0.5A, and at 250VAC you can switch up to 0.6A.

Power Relay



Adafruit Power Relay FeatherWing

$9.95
IN STOCK

ADD TO CART

This Wing has a *non-latching* type relay. Compared to our smaller mini relay FeatherWings, this one can handle a beefy 1200 Watts!

You can switch up to **10A** of resistive-load current at 120VAC, 5A at 240VAC. With inductive loads, about half that. Check the datasheet for the relay for the exact switching capacity, as it depends on type of load and voltage type and magnitude. This relay is good for handling fairly large devices, computers, TVs, small appliances and more.

Power Monitoring Wing

Your browser does not support the video tag. Adafruit INA219 FeatherWing

$7.95
IN STOCK

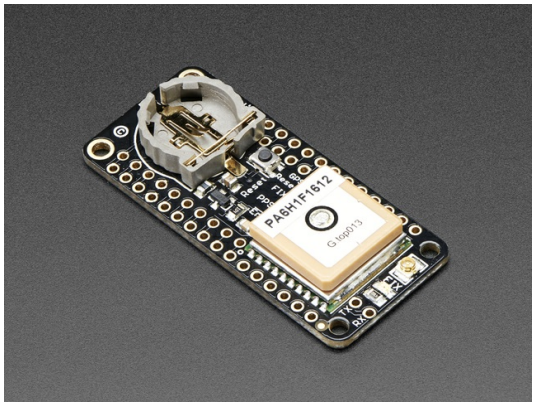
ADD TO CART

The **INA219 FeatherWing** makes power-monitoring problems a thing of the past. Instead of struggling with two multimeters, you can just use the handy INA219B chip on this breakout to both measure both the high side voltage and DC current draw over I2C with 1% precision. **Works with any and all Feathers!** Communicates over I2C so its super-simple to use, you can even change the I2C address to have up to 4 of these Wings on one Feather.

Many current-measuring devices are only good for *low side* measuring. That means that unless you want to get a battery involved, you have to stick the measurement resistor between the target ground and true ground. This can cause problems with circuits since electronics tend to not like it when the ground references change and move with varying current draw. This chip is much smarter - it can handle high side current measuring, up to +26VDC, even though it is powered with 3.3V. It will also report back that high side voltage, which is great for tracking battery life or solar panels.

A precision amplifier measures the voltage across the 0.1 ohm, 1% sense resistor. Since the amplifier maximum input difference is ±320mV this means it can measure up to ±3.2 Amps. With the internal 12 bit ADC, the resolution at ±3.2A range is 0.8mA. With the internal gain set at the minimum of div-8, the max current is ±400mA and the resolution is 0.1mA. Advanced hackers can remove the 0.1 ohm current sense resistor and replace it with their own to change the range (say a 0.01 ohm to measure up 32 Amps with a resolution of 8mA).

GPS Wing



Adafruit Ultimate GPS FeatherWing

$39.95
IN STOCK

ADD TO CART

This Wing is our only GPS/GNSS capable wing. It's great for adding location or time awareness to your project.

That said, it is the Wing with the most compatibility issues. That's because it uses UART serial to send/receive data. And many microcontrollers have only one UART and it's used for programming and debugging.

These are the Feathers that are not compatible:

- **ESP8266 Huzzah Feather**
- **nRF52 Bluefruit Feather**

All other Feathers have a free UART (mostly because they have native USB for upload/debug) or multiple UARTs, like the ESP32.

You can only have one GPS, but it can be used with any other Wing (since it's the only UART-using wing)

Antenna Reception

Like all GPS devices, the Wing needs a good antenna orientation to receive the tiny signals sent from space. The GPS Wing has a built in antenna, which can be used as long as the Feather is in a non-metallic enclosure and facing up.

You can use a uFL -> SMA and SMA antenna to add a big antenna for good reception when in an enclosure.

SMA to uFL/u.FL/IPX/IPEX RF Adapter Cable

$3.95
IN STOCK

ADD TO CART

GPS Antenna - External Active Antenna - 3-5V 28dB 5 Meter SMA

$14.95
IN STOCK

ADD TO CART

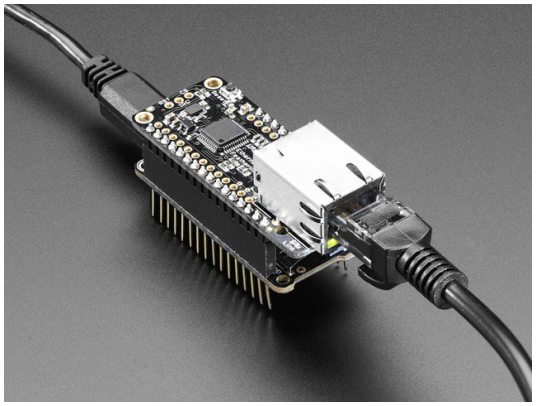
For best reception and performance, adding a backup battery will keep the GPS from having to download the 'Almanac' every time the Feather is turned on. Pick up a CR1220 battery and install it!

CR1220 12mm Diameter - 3V Lithium Coin Cell Battery

$0.95
IN STOCK

ADD TO CART

Ethernet Wing



Adafruit Ethernet FeatherWing

$19.95
IN STOCK

ADD TO CART

Ethernet is a reliable standard for Internet connectivity. If you don't need Wireless (and don't want the headache of wireless either), then Ethernet can be a very good option.

No antennas, no SSIDs, no passwords, plug it in and forget about it.

While there are a few Feathers that have built-in 'Ethernet support' they all need a PHY interface anyways and it's about the same price as the W5500 chipset we use. So we just went with this, which is a well-known (or, at least, infamous) chipset. And, yes, you can use it with our WiFi Feathers to create a Feather with dual-connectivity options (or a very slow and silly WiFi bridge)

This Wing uses the SPI pins and a CS pin, you can cut the jumper to re-assign the CS pin. You can use with other SPI-based wings (displays, music, radio, datalogging, etc) just make sure there's no pin conflict for the CS pin.

This Wing does not support active Power-over-Ethernet (PoE), but you can easily hook up a passive PoE configuration with a passive injector, a 5V power adapter, and a 2.1mm to MicroUSB cable. See the tutorial page for more details (https://adafru.it/wiC)

RTC & Datalogging Wings

While all the Feathers have a LiPo battery backup capability, they don't always have a true "Real Time Clock" that can keep the correct time.

If you have a WiFi or Cellular Feather you may be able to get away with not having a true RTC because you can connect to the network (say, NTP) and query the time. But, even then, there's a chance you'll be off-grid and need to know the time and date.

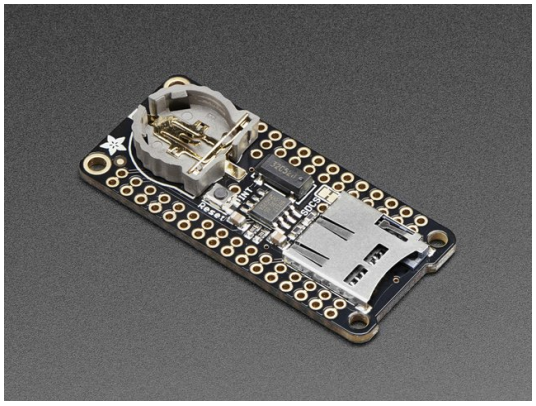
Thus we have two options for adding an RTC. Both RTC's are I2C, and they share the same address. So you can only have one RTC, but you can match them up with any other Wings. If using the SD card on the Adalogger, just make sure the CS pin is not shared.

Both Wings work with all Feathers. A CR1220 coin cell is required and not included (to make air-shipping abroad easier they are sold separately)

CR1220 12mm Diameter - 3V Lithium Coin Cell Battery

$0.95
IN STOCK
ADD TO CART

Adalogger + RTC Wing



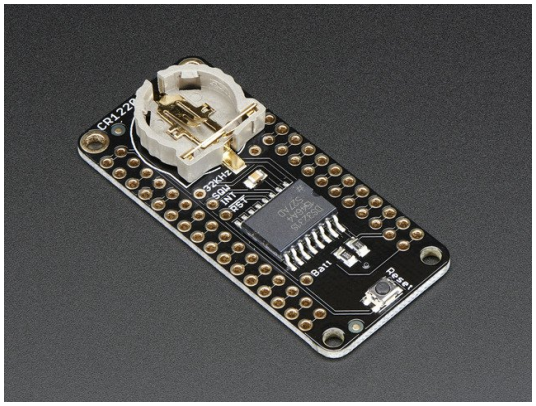
Adalogger FeatherWing - RTC + SD Add-on For All Feather Boards

$8.95
IN STOCK
ADD TO CART

This FeatherWing will make it real easy to add datalogging to any of our existing Feathers. You get both an I2C real time clock (PCF8523) with 32KHz crystal and battery backup, and a microSD socket that connects to the SPI port pins (+ extra pin for CS).

If you don't need to use the SD card socket, leave it empty and the SPI + CS pins can be used for something else.

Precision RTC Wing



DS3231 Precision RTC FeatherWing - RTC Add-on For Feather Boards

$13.95
IN STOCK

ADD TO CART

The PCF8523 used in the Adalogger has an external 32kHz timing crystal that is used to keep time with low current draw. And that's all well and good, but those crystals have slight drift, particularly when the temperature changes (the temperature changes the oscillation frequency very very very slightly but it does add up!)

The DS3231 RTC is in a beefy package because the crystal is inside the chip! And right next to the integrated crystal is a temperature sensor. That sensor compensates for the frequency changes by adding or removing clock ticks so that the timekeeping stays on schedule.

You'll get much higher precision with this RTC, at an additional cost. And there is no space for an SD card holder.

Music Wings

If you'd like to create a music-playing project, adding a Music Maker Wing will make it easy. These feature the VS1053 chipset which can do MP3, Ogg Vorbis, WAVE playback. They also have a MIDI synth inside that can be controlled over UART.

Both Music Maker Wings have the same circuitry/code but one has a headphone jack and the other has a 2 Watt audio amplifier. If you want to connect speakers directly, go with the amp. If you want to use headphones or connect to a powered stereo, the headphone one will work fine.

Main SPI Mode compatibility

There's a lot going on with these Wings, and they have some incompatibilities:

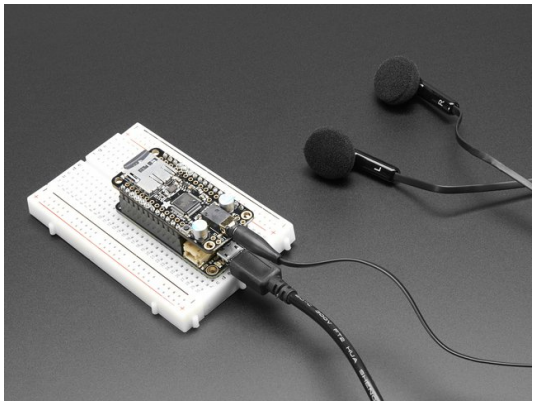
If you are only playing audio (MP3/Ogg/Wave), then these Wings are compatible with all Feathers.

But there are *a lot* of pins used: SPI + 3 for the MP3 chip and +1 for the SD card. MP3 playing works best with an interrupt, so those pins can be used at any time. We don't even recommend using this with Feathers that want to access SPI at their leisure, such as the Adalogger, non-nRF52 Bluefruit, or Radio. It *is possible* but use care to make sure you aren't using the two SPI devices at once.

We only recommend using this Wing along-side I2C-based FeatherWings to make sure there are no conflicts

MIDI Synth Mode compatibility

If you want to use the UART MIDI synth (which is not the same mode as the MP3/Ogg/Wave player), you'll need to use the UART TX pin. For the **nRF52** or **ESP8266** Feathers, you may need to remove the 'Wing during programming. And, you'll end up re-using the main Serial console for MIDI which can be confusing.



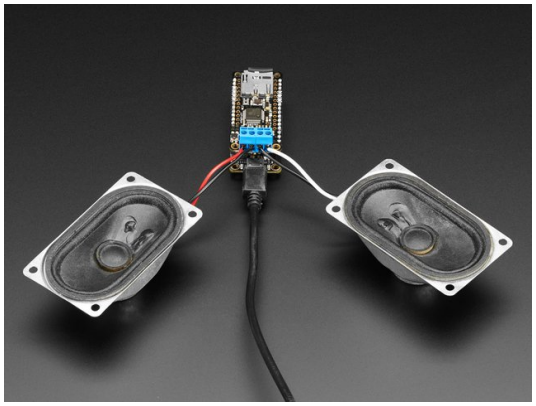
Adafruit Music Maker FeatherWing - MP3 OGG WAV MIDI Synth Player

$19.95
IN STOCK

ADD TO CART

The Feather has the MP3 decoder chip and an SD card socket. You can stream MP3 directly to the chip if you like (say from a WiFi connection or SPI Flash if you have an Express) or read it from the SD card.

The headphone-out version has blocking capacitors so you can connect it to headphones or a powered stereo system.



Music Maker FeatherWing w/ Amp - MP3 OGG WAV MIDI Synth Player

$24.95
IN STOCK

ADD TO CART

The amplifier-out Wing is just like the headphone Wing but with a stereo Class D audio amplifier. This amp can draw quite a bit of current, 1A easily, if you're really bumping out some tunes! So make sure you have a beefy battery and/or 5V 2A wall adapter

5V 2.5A Switching Power Supply with 20AWG MicroUSB Cable

$7.50
IN STOCK

ADD TO CART

5V 2A Power Supply w/ 20AWG 6' MicroUSB Cable - International

$14.95
OUT OF STOCK

OUT OF STOCK

Game and Prop Wings

Feather is a perfect form factor for handheld gaming, to put inside wearables and props.

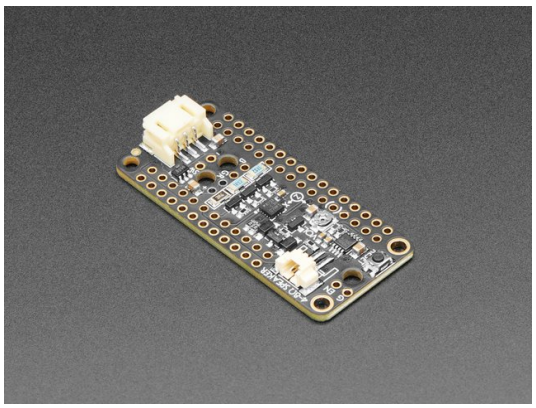
Here are FeatherWings which may be added to a Feather to help your creative side.

Your browser does not support the video tag.

Adafruit Joy FeatherWing for all Feathers

$9.95
IN STOCK
ADD TO CART

Add a joystick and buttons with the cute Joy FeatherWing. This FeatherWing has a 2-axis joystick and 5 momentary buttons (4 large and 1 small) so you can turn your feather board into a tiny game controller. This wing communicates with your host microcontroller over I2C so it's easy to use and doesn't take up any of your precious analog or digital pins. There is also an optional **interrupt pin** that can alert your feather when a button has been pressed or released to free up processor time for other tasks.



Adafruit Prop-Maker FeatherWing

$9.95
IN STOCK
ADD TO CART

The Prop-Maker FeatherWing will unlock the prop-maker inside all of us, with tons of stuff packed in to make sabers & swords, props, toys, cosplay pieces, and more.

We looked at hundreds of prop builds, and thought about what would make for a great low-cost (but well-designed) add-on for our Feather boards. Here's what we came up with:

- **Snap-in NeoPixel port -** With a 3-pin JST connector, you can plug in one of our JST-wired NeoPixel strips directly (https://adafru.it/Cup), or use a 3-pin JST connector to wire up your favorite shape of addressable NeoPixel LEDs (https://adafru.it/DBh). This port provides high current drive from either the Feather Lipoly or USB port, whichever is higher. A level shifter gives you a clean voltage signal to reduce glitchiness no matter what chip you're using
- **3W RGB LED drivers** - 3 high current MOSFETs will let you connect a 3W RGB LED for powerful eye-blasting glory (https://adafru.it/CXi). For most Feathers, the 3 pins are PWM capable so you can generate any color you

like. Available as pin breakouts plus strain-relief holes

- **Triple-Axis Accelerometer with Tap Detection** - The LIS3DH is our favorite accelerometer, you can use this for detection motion, tilt or taps. Here's an example of a light saber that makes sounds when swung or hit. (https://adafru.it/DBi) We have code for this chip in both Arduino and CircuitPython.
- **Class D Audio Amplifier** - Drive a 8Ω 1 Watt speaker or 4Ω 3W speaker for sound effects. Plug and play with our cute and slim oval speaker (https://adafru.it/CEv), or connect a picoblade cable (https://adafru.it/CVi) for your favorite speaker. For use only with Feathers that have analog audio out such as the Feather M0 Express and M4 series.
- **Low power mode!** The power system for the RGB LED, NeoPixels and speaker amplifier can be controlled by a pin to cut power to them, so you have lower power usage when the prop is in sleep or off mode (but can wake up fast by listening to the button press or accelerometer data). When the power pin is set low, the current draw for just the wing is under 1mA and no there's current draw from any attached NeoPixels - normally they're about 1mA even when not lit.
- Breakouts plus strain-relief hole for the enable pin and ground (for a mechanical switch that will power down the whole board)
- Breakouts plus strain-relief holes for an external switch pin and ground (for a mechanical mode button)

Please note: A few of the onboard hardware elements use PWM and analog output so we recommend the **Feather M0 Express** or **Feather M4** series, they'll work best with this wing and let you make the most of it.

Camera Wing

The current FeatherWIng lineup has one camera wing:

Your browser does not support the video tag. Adafruit AMG8833 IR Thermal Camera FeatherWing

$39.95
IN STOCK
ADD TO CART

Thermal Camera FeatherWing: thanks to the Panasonic AMG8833 8x8 GridEYE sensor, it adds heat-vision to *any* Feather main board. Using our Feather Stacking Headers (http://adafru.it/2830) or Feather Female Headers (http://adafru.it/2886) you can connect a FeatherWing on top of your Feather board and let the board take flight!

This sensor from Panasonic is an 8x8 array of IR thermal sensors. When connected to your Feather it will return an array of 64 individual infrared temperature readings over I2C. It's like those fancy thermal cameras, but compact and simple enough for easy integration.

This part will measure temperatures ranging from **0°C to 80°C (32°F to 176°F)** with an accuracy of +- 2.5°C (4.5°F). It can detect a human from a distance of up to 7 meters (23) feet. With a maximum frame rate of 10Hz, It's perfect for creating your own human detector or mini thermal camera. We have an easy-to use Arduino and CircuitPython code so you can get started fast. The sensor communicates over I2C. If you have a fast Feather like the ESP8266, ESP32 or Teensy, you can interpolate the 8x8 grid and get some pretty nice results! (The video above shows a peace-sign finger demo using a Teensy Feather and 24x24 interpolation)

The AMG8833 is the next generation of 8x8 thermal IR sensors from Panasonic, and offers higher performance than its predecessor the AMG8831. The sensor only supports I2C, and has a configurable interrupt pin that can fire when any individual pixel goes above or below a threshold that you set.

Community Feathers & Wings

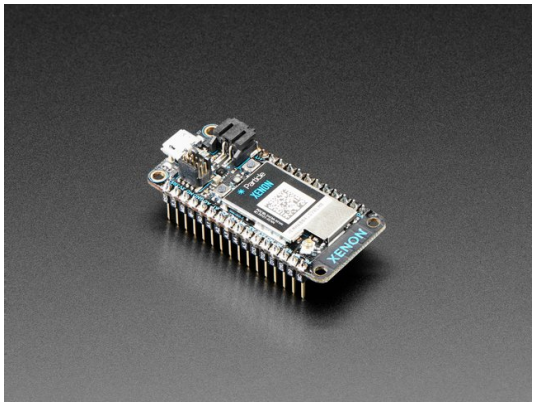
Adafruit has created dozens and dozens of Feathers and Wings, but there's a chance we haven't made the exact Wing you want. And that's totally fine! We've specifically designed the Feather system so that you can create your own custom circuitry, and move it from Feather to Feather.

Here are just a *few* examples of Feathers and Wings created by the community.

If you'd like to make your *own* Feather or Wing, check out our GitHub account and search for Feather and PCB to get open source designs that you can use to base your own design on (https://adafru.it/woC)

Community Feathers

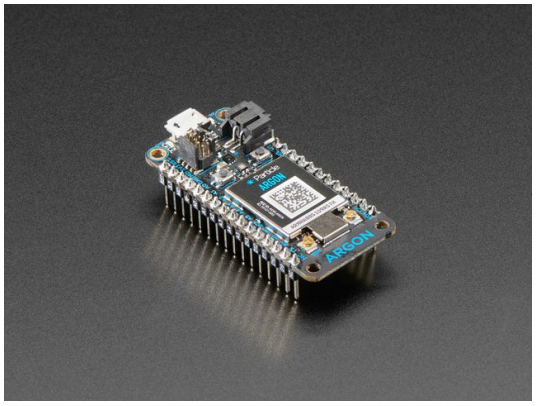
Particle has introduced three processor boards with built-in communications:



Particle Xenon - nRF52840 with BLE and Mesh

$19.95
IN STOCK

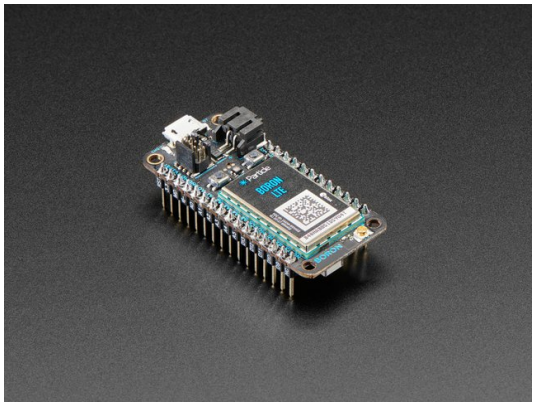
ADD TO CART



Particle Argon - nRF52840 with Mesh and WiFi

$27.50
IN STOCK

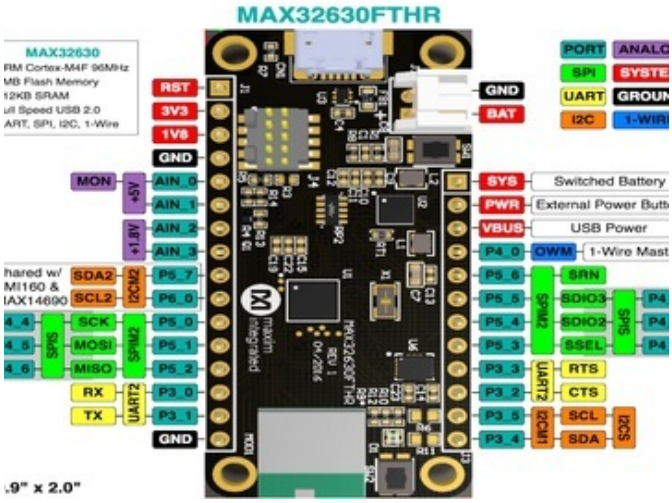
ADD TO CART



Particle Boron LTE - nRF52840 with Mesh and LTE Cellular Modem

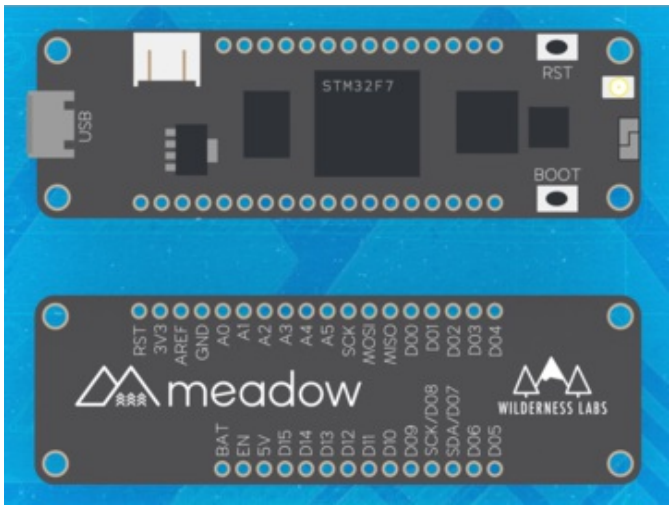
$55.00
IN STOCK

ADD TO CART



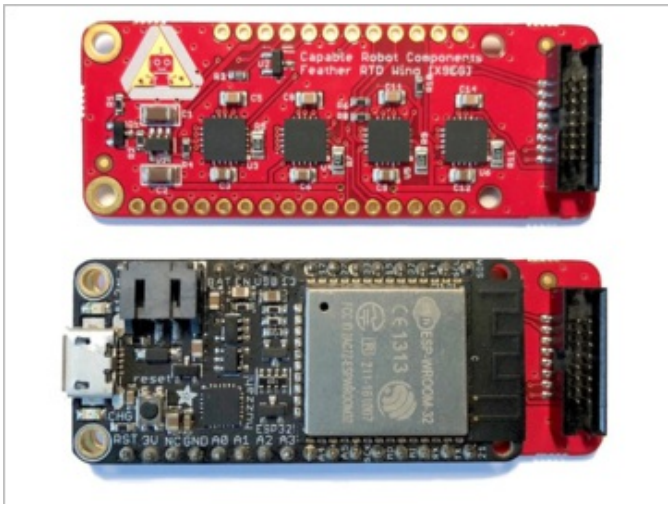
Maxim developed their latest IoT platform dev boards to be Feather compatible. Given the pinout, it should be compatible with almost all Wings

- MAX32620FTHR (https://adafru.it/DOz)
- MAX32630FTHR (https://adafru.it/DOA)



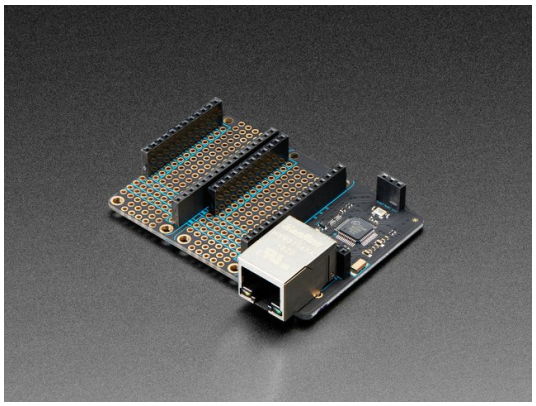
The Wilderness Labs Meadow (https://adafru.it/DOB) board contains a STM32F7 processor running .NET with WiFi and Bluetooth.

Some Community Wings



Capable Robot Components supplies the SenseTemp (https://adafru.it/DOC) board in FeatherWing format.

SenseTemp is an open source, four-channel temperature sensor designed for instrumenting electronics. It uses extremely accurate platinum resistive temperature detector (RTD) elements which are small enough to place directly on ICs, heatsinks, and other points of interest on an electronic circuit board.

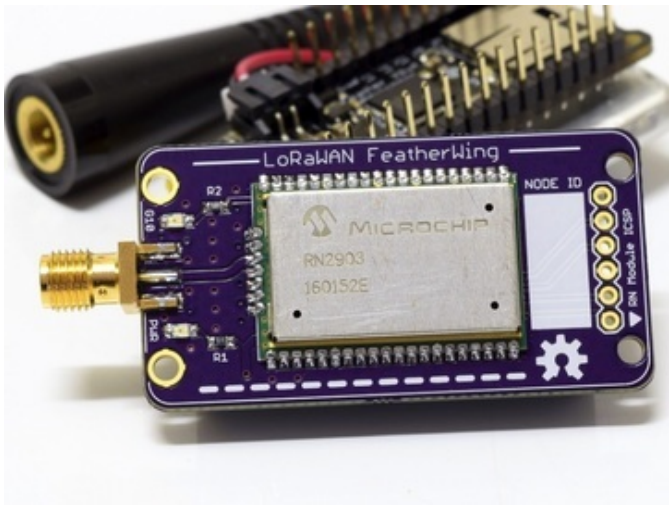


Particle Ethernet FeatherWing

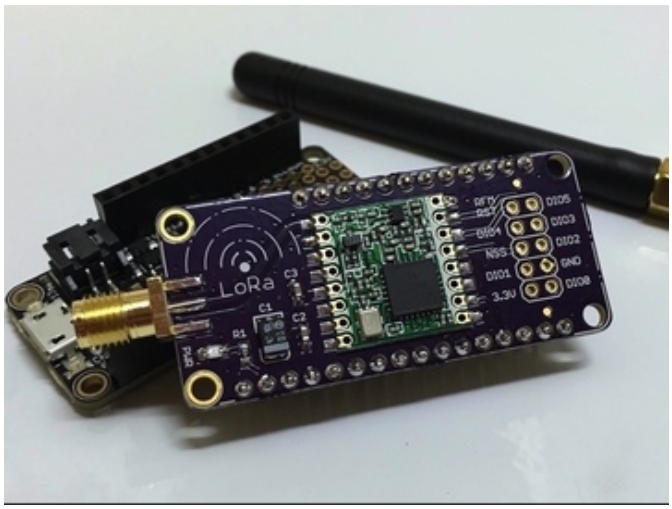
$20.00
IN STOCK

ADD TO CART

The Particle Classic Adapter (https://adafru.it/DOD) allows classic Particle accessories to connect to the Feather formatted Particle boards.

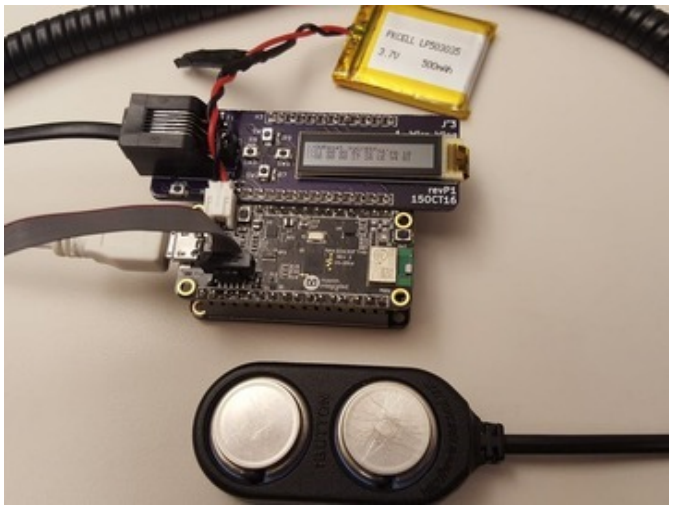


Dan Watson made a lovely LoRaWAN FeatherWing, check it out here (https://adafru.it/woE)



Dan Watson also has a RFM-style module breakout (https://adafru.it/DOF) FeatherWing, when you don't mind implementing LoRaWAN in-chip

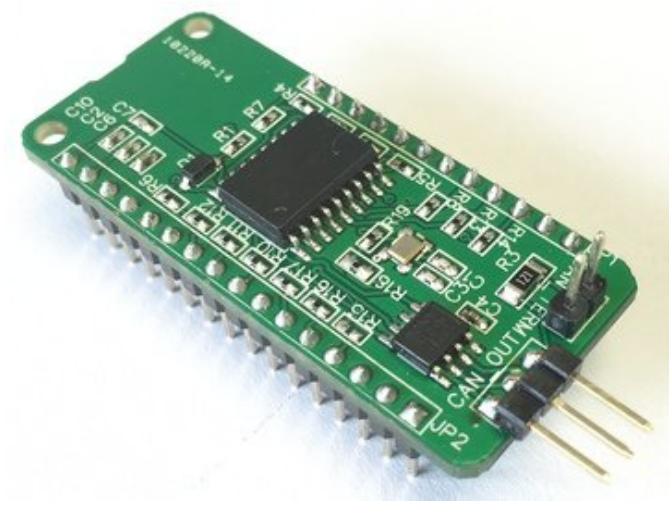
Radomir Dopieralski (https://adafru.it/wpa) made a cool mini-game FeatherWing with a D-Pad, 8x8 Mini LED with PWM driver! (https://adafru.it/wpb)



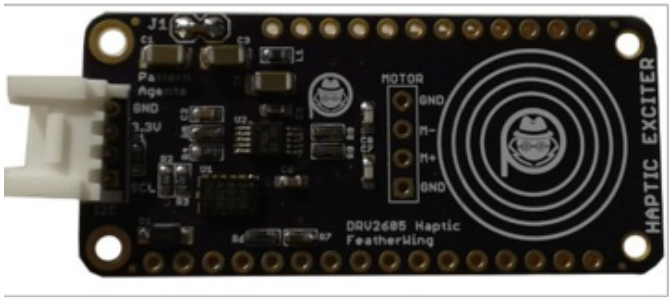
Justin Jordan created a Wing for his Maxim Feather that adds lots of 1-Wire sensor support. Check it out at Hackster (https://adafru.it/wpc)

Tisham Dhar made a cool energy-monitoring FeatherWing using the ADS1115 16-bit ADC (https://adafru.it/wpd)

Tisham also has for sale a ATM90E26 FeatherWing (https://adafru.it/DOE) for more advanced power monitoring available for purchase at Tindie.



Armin created a CAN FeatherWing using the MCP2515 CAN controller with an 3.3V CAN transceiver (https://adafru.it/wpA)



PatternAgents Agent-DRV2605 (https://adafru.it/DSh) FeatherWing contains a TI DRV2605L Haptic Driver and ADI ADXL345 Accelerometer.

The Agent-DA7280 (https://adafru.it/DSh) has a DialogSemi DA7280L Haptic Driver and ADI ADXL345 Accelerometer.