Measuring Battery

If you're running off of a battery, chances are you wanna know what the voltage is at! That way you can tell when the battery needs recharging. Lipoly batteries are 'maxed out' at 4.2V and stick around 3.7V for much of the battery life, then slowly sink down to 3.2V or so before the protection circuitry cuts it off. By measuring the voltage you can quickly tell when you're heading below 3.7V

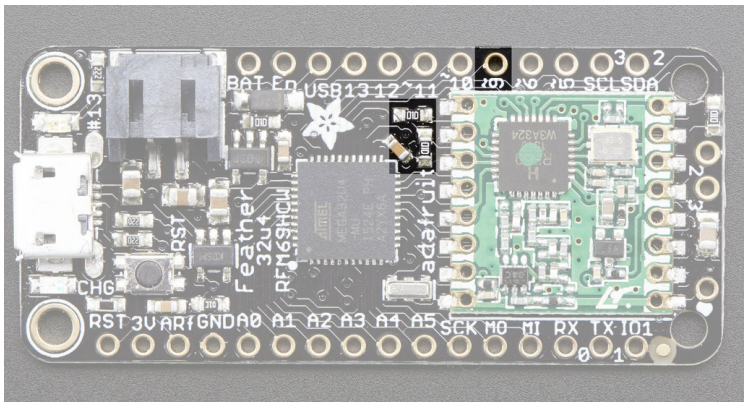
To make this easy we stuck a double-100K resistor divider on the **BAT** pin, and connected it to **D9** (a.k.a analog #7 **A7**). You can read this pin's voltage, then double it, to get the battery voltage.

Download: file
Copy Code

```
#define VBATPIN A9

float measuredvbat = analogRead(VBATPIN);
measuredvbat *= 2;    // we divided by 2, so multiply back
measuredvbat *= 3.3;  // Multiply by 3.3V, our reference voltage
measuredvbat /= 1024; // convert to voltage
Serial.print("VBat: " ); Serial.println(measuredvbat);
```

This voltage will 'float' at 4.2V when no battery is plugged in, due to the lipoly charger output, so its not a good way to detect if a battery is plugged in or not (there is no simple way to detect if a battery is plugged in)



Alternative Power Options

The two primary ways for powering a feather are a 3.7/4.2V LiPo battery plugged into the JST port *or* a USB power cable.

If you need other ways to power the Feather, here's what we recommend:

- For permanent installations, a 5V 1A USB wall adapter will let you plug in a USB cable for reliable power
- For mobile use, where you don't want a LiPoly, use a USB battery pack!
- If you have a higher voltage power supply, use a 5V buck converter and wire it to a USB cable's 5V and GND input

Here's what you cannot do:

- **Do not use alkaline or NiMH batteries** and connect to the battery port - this will destroy the LiPoly charger and there's no way to disable the charger
- **Do not use 7.4V RC batteries on the battery port** - this will destroy the board

The Feather *is not designed for external power supplies* - this is a design decision to make the board compact and low cost. It is not recommended, but technically possible:

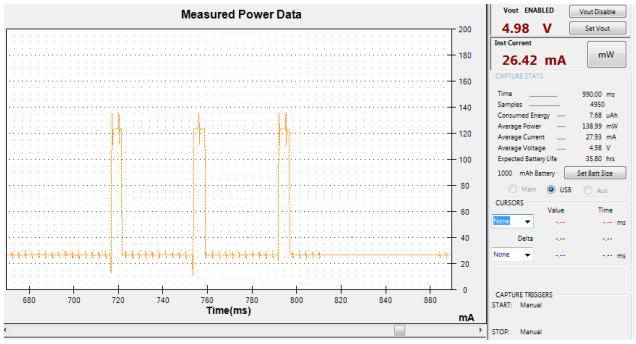
- **Connect an external 3.3V power supply to the 3V and GND pins.** Not recommended, this may cause unexpected behavior and the **EN** pin will no longer. Also this doesn't provide power on **BAT** or **USB** and some Feathers/Wings use those pins for high current usages. You may end up damaging your Feather.
- **Connect an external 5V power supply to the USB and GND pins.** Not recommended, this may cause unexpected behavior when plugging in the USB port because you will be back-powering the USB port, which *could* confuse or damage your computer.

Radio Power Draw

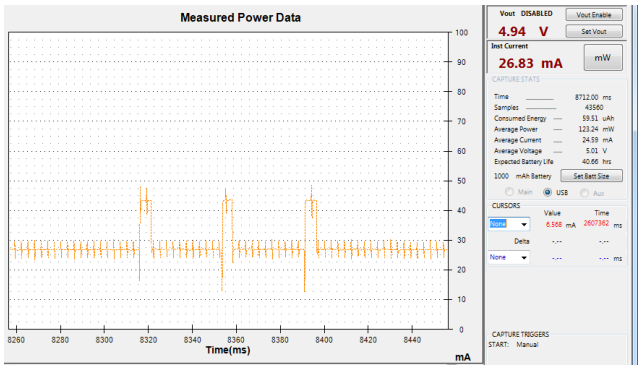
You can select the power output you want via software, more power equals more range but of course, uses more of your battery.

For example, here is the feather 32u4 with RFM69HCW set up for +20dBm power, transmitting a data payload of 20 bytes

The ~25mA quiescent current is the current draw for listening (~15mA) plus ~10mA for the microcontroller. This can be reduce to amost nothing with proper sleep modes and not putting the module in active listen mode!

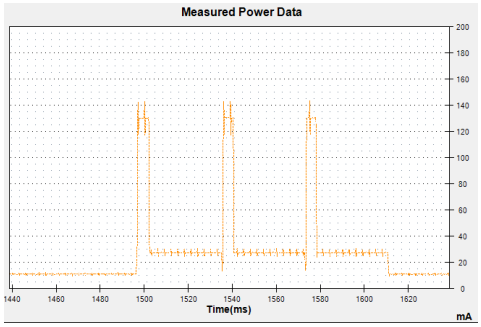


Here is a transmit with `radio.setPowerLevel(0)` to set +5dBm power



You still have the 25mA average, but during transmit, it only uses another 20mA not 100mA

If you put the radio to sleep after transmitting, rather than just sitting in receive mode, you can save more current, after transmit is complete, the average current drops to ~10mA which is just for the microcontroller



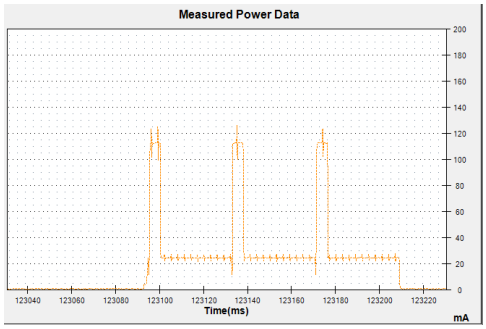
If you want to reduce even more power, use the Adafruit Sleepdog library by installing and adding #include "Adafruit_SleepyDog.h" at the top of your sketch and replace

```
delay(1000);
```

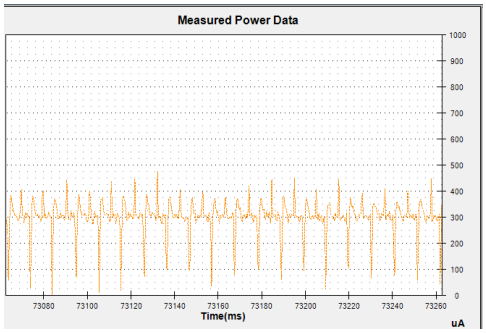
with

```
radio.sleep();
Watchdog.sleep(1000);
```

To put the chip into ultra-low-power mode. Note that USB will disconnect so do this after you have done all your debugging!



During the super sleepy mode you're using only 300uA (0.3mA)!

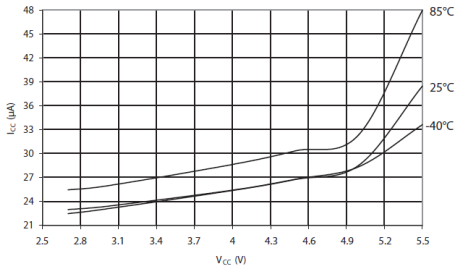


While its not easy to get the exact numbers for all of what comprise the 300uA there are a few quiescent current items on the Feather 32u4:

- 2 x 100K resistors for VBAT measurement = **25uA**
- AP2112K 3.3V regulator = **55uA**
- MCP73871 batt charger = **up to 100uA** even when no battery is connected

The rest is probably the Atmega32u4 peripherals including the brown-out detect and bandgap circuitry, ceramic oscillator, etc. According to the datasheet, with the watchdog and BrownOutDetect enabled, the lowest possible current is **~30uA** (at 5V which is what we're testing at)

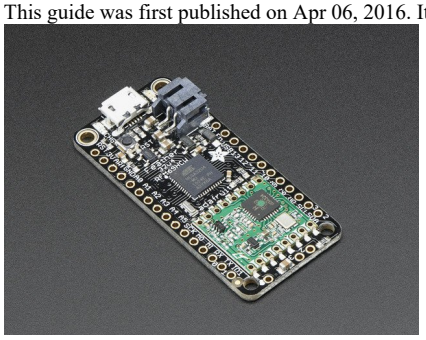
Figure 30-12. Power-down Supply Current vs. V_{CC} (WDT Enabled, BOD EN)

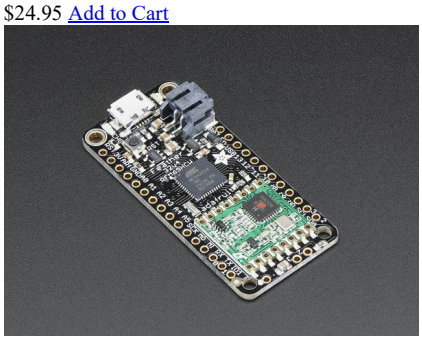


ENable pin

If you'd like to turn off the 3.3V regulator, you can do that with the **EN**(able) pin. Simply tie this pin to **Ground** and it will disable the 3V regulator. The **BAT** and **USB** pins will still be powered

ANTENNA OPTIONS ARDUINO IDE SETUP
This guide was first published on Apr 06, 2016. It was last updated on Apr 06, 2016. This page (Power Management) was last updated on Jan 23, 2019.


Adafruit Feather 32u4 RFM69HCW Packet Radio - 868 or 915 MHz
$24.95 Add to Cart


Adafruit Feather 32u4 with RFM69HCW Packet Radio - 433MHz
$24.95 Add to Cart