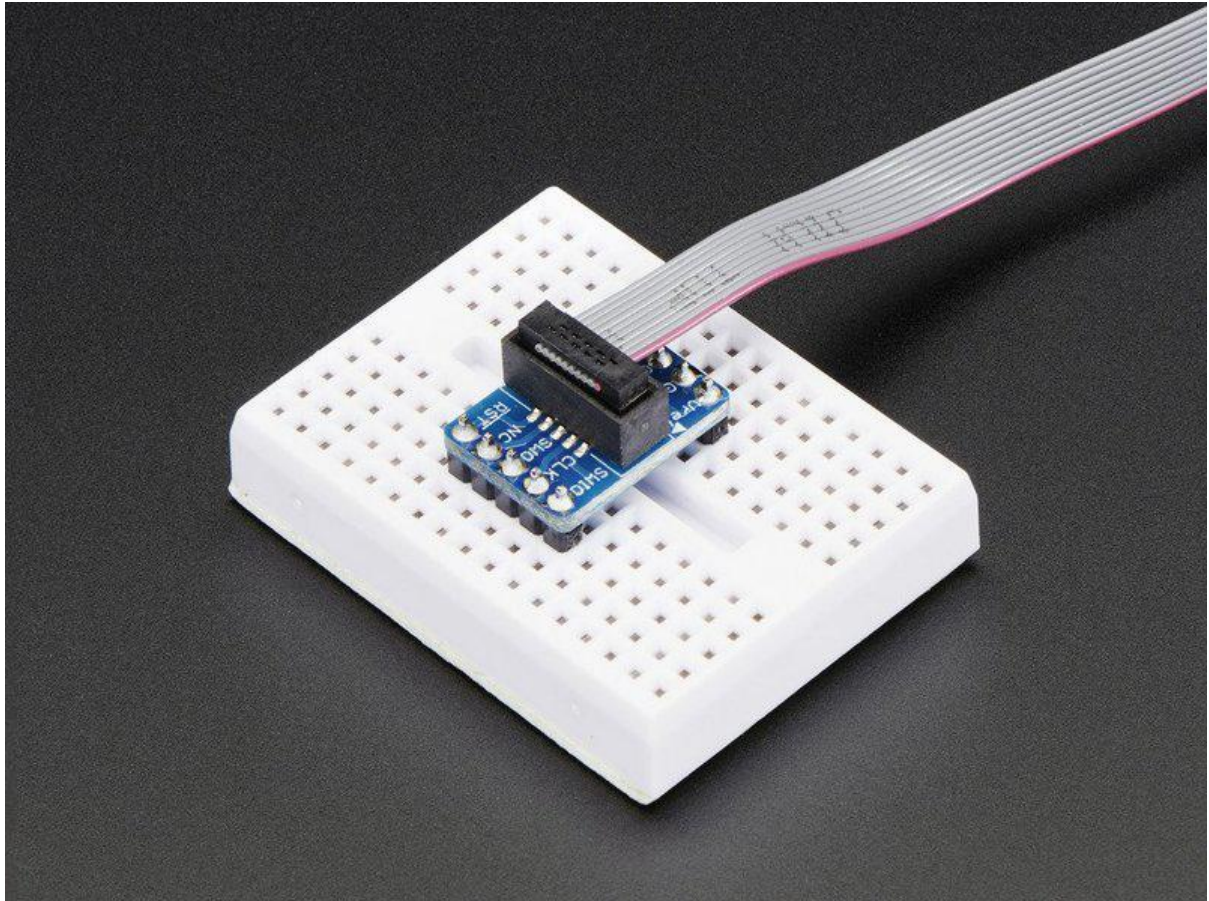




Programming an M0 using an Arduino

Created by Dean Miller



<https://learn.adafruit.com/programming-an-m0-using-an-arduino>

Last updated on 2023-08-29 03:25:34 PM EDT

Table of Contents

Overview

3

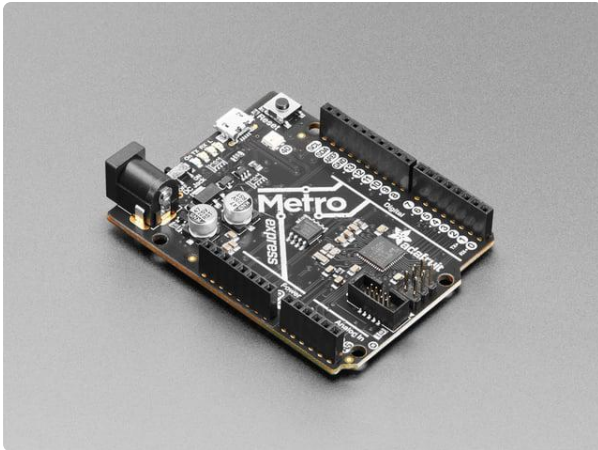
- [Wiring](#)
- [Load Firmware](#)
- [Download Adafruit_DAP library](#)
- [Load Example](#)
- [Support & More Details & Other Chips](#)

Overview

If you've ever had to ship a bunch of pre-programmed ARM based microcontrollers (and we have), you know that it can be a bulky process. The traditional way to do it requires a computer and a J-Link or some other type of programmer.

However, most Atmel ARM cortex microcontrollers can be programmed via the Debug Access Port (DAP). This is made easy Thanks to the excellent [free-dap project](#) () by Alex Taradov.

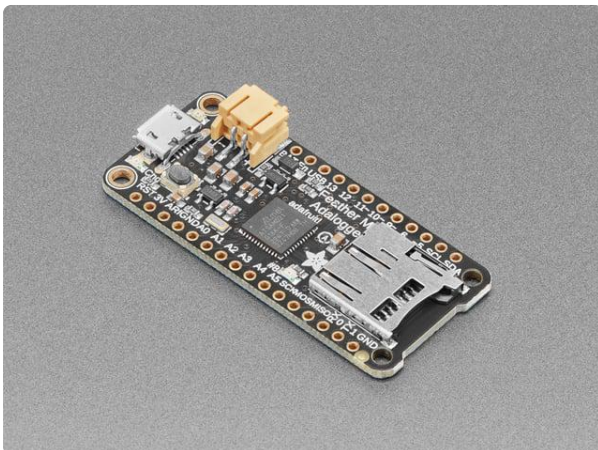
What we've done is package up both the host and device components of free-dap into something that's Arduino-compatible, so you can use an ATSAM21 based board such as the Feather M0 or Metro M0 to program boards quickly and easily with no host system.



[Adafruit METRO M0 Express - designed for CircuitPython](#)

Metro is our series of microcontroller boards for use with the Arduino IDE. This new Metro M0 Express board looks a whole lot like our

<https://www.adafruit.com/product/3505>



[Adafruit Feather M0 Adalogger](#)

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

<https://www.adafruit.com/product/2796>

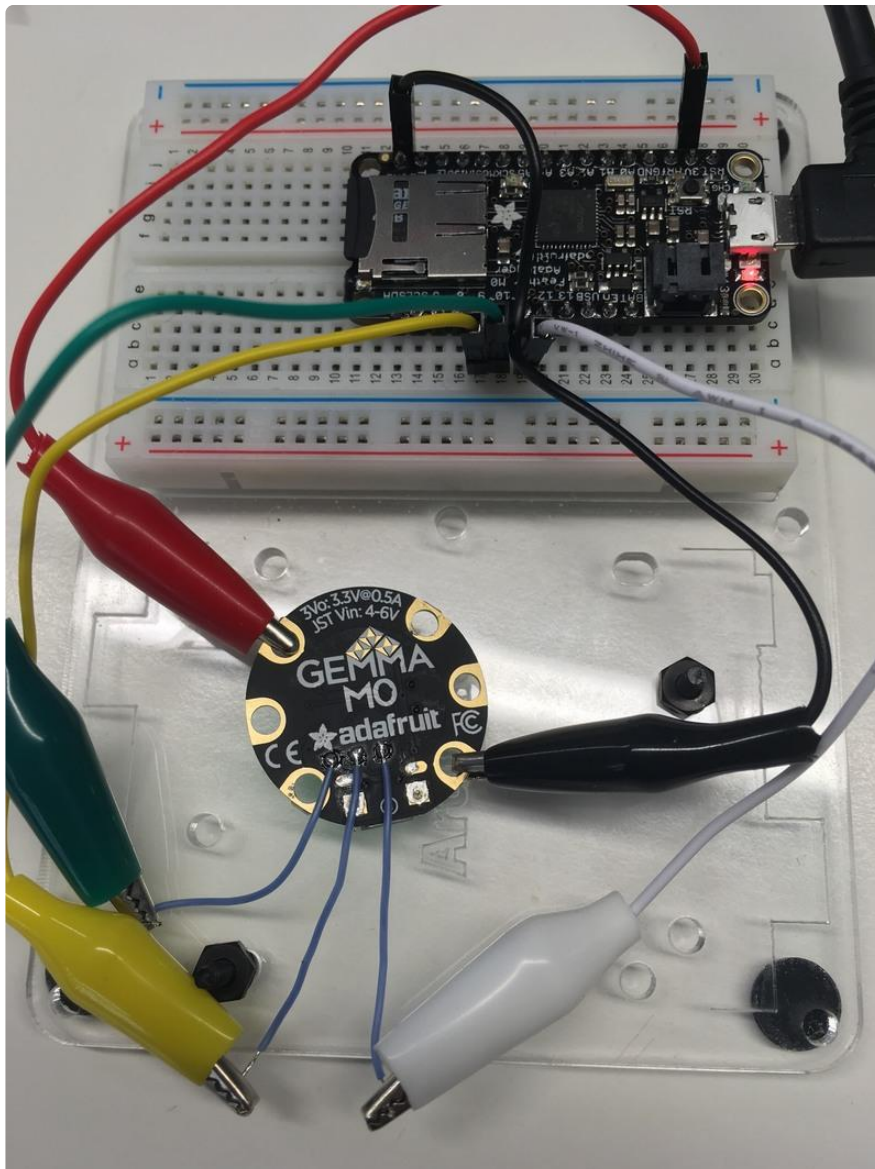
Note that an ATSAM21 based board must be used as the programmer. AVR based boards like the Feather 32u4 will not work.

Wiring

In this tutorial we will be using a Feather M0 Adalogger to program a Gemma M0 from an SD card.

First we will need to solder wires to the exposed SWD pads on the back of the Gemma M0 so we can access the DAP.

- connect the pad closest to the edge of the Gemma (SWDIO) to pin 10 of the Feather
- connect the middle pad on the Gemma (SWCLK) to pin 9 of the Feather
- connect the right-most pad on the Gemma (nRESET) to pin 11 of the Feather
- connect 3vo on the Gemma to 3v on the Feather
- connect GND on the Gemma to GND on the Feather



Load Firmware

Place the binary file you would like to program onto the Gemma on a microSD card and title it fw.bin.

Make sure you are writing a .bin file to the microcontroller. A .hex file will not work with this example.

Download Adafruit_DAP library

To begin reading sensor data, you will need to download Adafruit_DAP from our github repository. You can do that by visiting the github repo and manually downloading or, easier, just click this button to download the zip

Download Adafruit DAP library

Rename the uncompressed folder Adafruit_DAP and check that the Adafruit_DAP folder contains Adafruit_DAP .cpp and Adafruit_DAP .h as well as a few other source files.

Place the Adafruit_DAP library folder your arduinosketchfolder/libraries/ folder. You may need to create the libraries subfolder if its your first library. Restart the IDE.

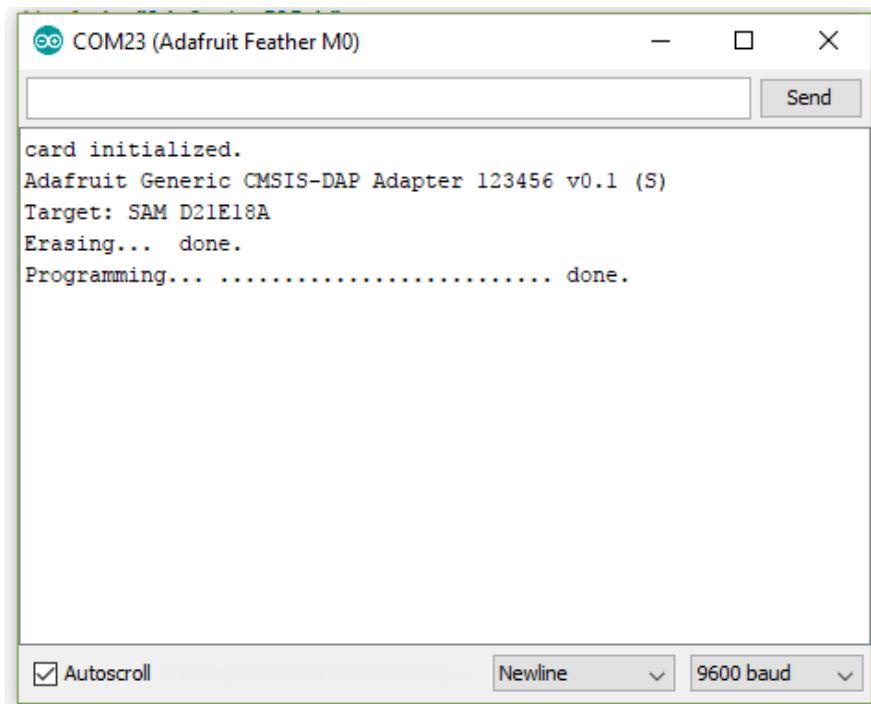
We also have a great tutorial on Arduino library installation at:

[http://learn.adafruit.com/adafruit-all-about-arduino-libraries-install-use \(\)](http://learn.adafruit.com/adafruit-all-about-arduino-libraries-install-use)

Load Example

Open up File->Examples->Adafruit_DAP->flash_from_SD and upload to your Feather wired up to the Gemma.

Once uploaded to your Feather, open up the serial console at 9600 baud speed to see the output. It will connect to the gemma and program it. You should see something like this:



And you're done! The Gemma is now running the code that was in the fw.bin file you placed on the SD card.

Support & More Details & Other Chips

You can also do things like get/set the user NVM (fuses), verify/read firmware on chip and run a CRC on the firmware through SWD which is much faster than verifying each byte. Check the `SWD_standalone_core` example for code snippets you can use to do the above.

This is a library we designed for internal use. Thus, it comes with absolutely no support whatsoever! It's something we thought people could find handy! There's no documentation beyond the examples, but it does work pretty well :)

Right now we only have commands for the ATSAMd series. [We'd be really into having other people add the SWD command magic for other chips, so if you do adapt this code for your favorite Cortex, please submit a pull request! \(\)](#)