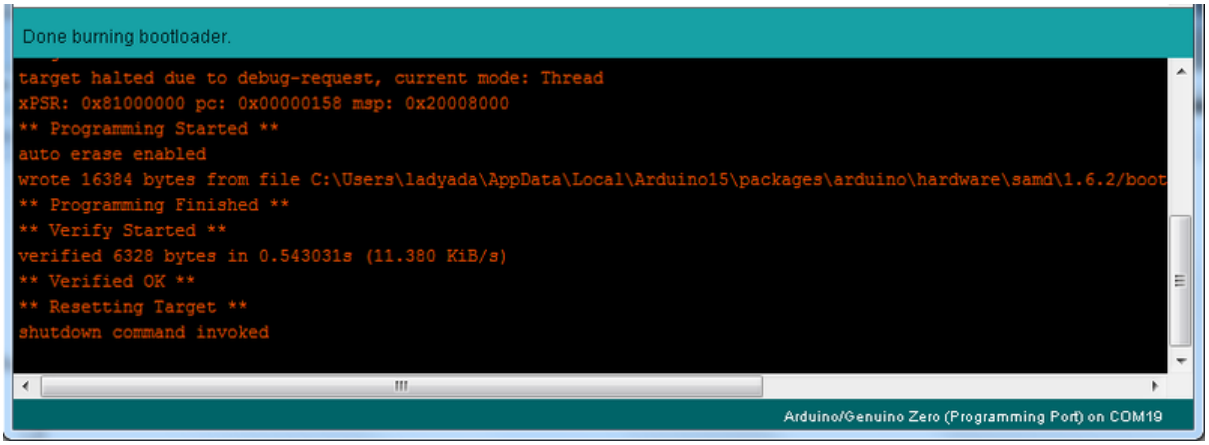
Compiling ATSAMD21 Bootloader

Created by lady ada



https://learn.adafruit.com/compiling-m0-atsamd21-bootloader

Last updated on 2024-06-03 01:53:04 PM EDT

Table of Contents

Compile

Note that this tutorial is for 'unlocked' fresh ATSAM chips. If you have a pre-programmed Feather M0 or Zero you may need to unlock the bootloader bits, we don't have a tutorial on doing that

WE DO NOT PROVIDE ANY SUPPORT FOR THIS TUTORIAL OR, IN GENERAL, HOW TO PROGRAM BOOTLOADERS OR ERASE CHIPS.

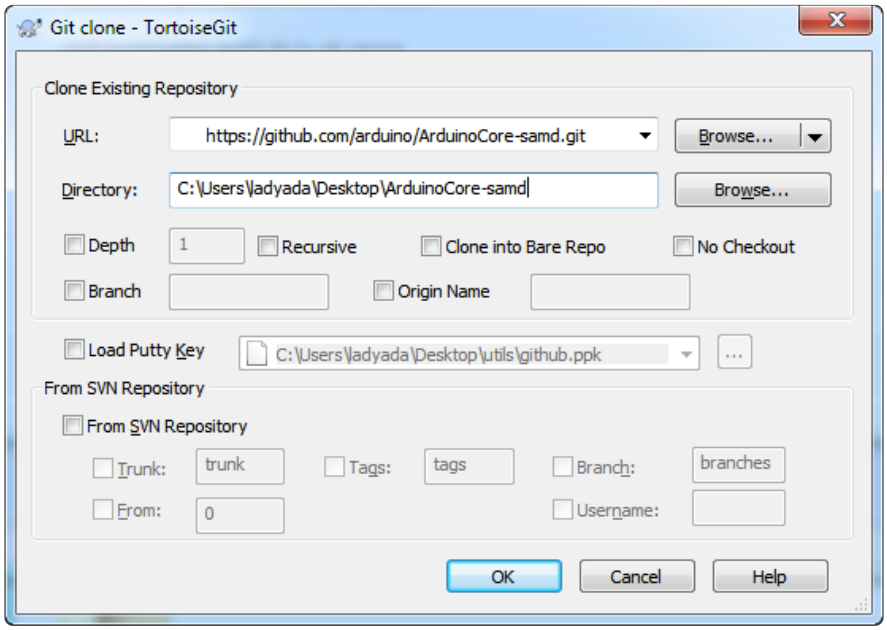
This isn't an in-depth tutorial, just some notes I took since I was asked a few times how to modify/compile the bossac bootloader!

Download Latest Bootloader Code

The latest Arduino SAMD core is at https://github.com/arduino/ArduinoCore-samd (https://adafru.it/mdi)

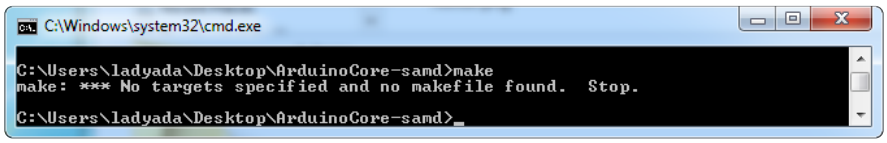
You can download a zip of the current snapshot (https://adafru.it/mdj) or you can use git to clone the repository

Its better to do this than compile in the board-support-package location (e.g. C:\user\blah\local\arduino15) because you'll lose your changes if you update the BSP!

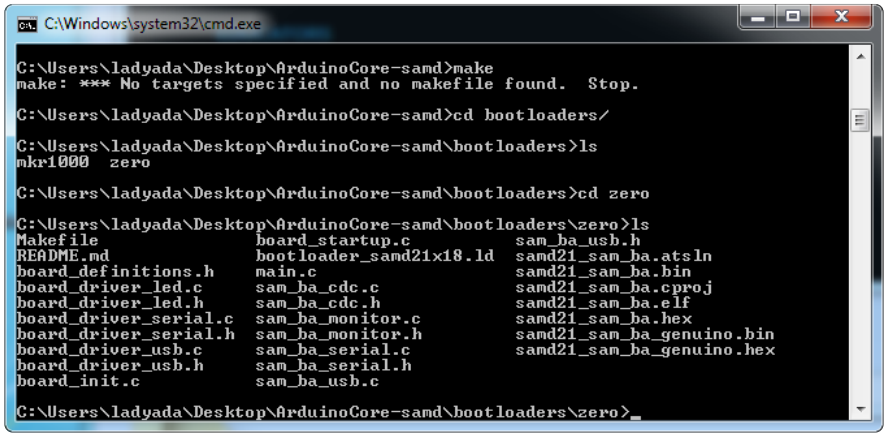


You'll need to have some basic command line compilation tools already installed like **make** - such as MSys or Cygwin (which we discuss here (https://adafru.it/mdk)) or install make as a binary (https://adafru.it/mdl)

Just make sure when you run **make** in a command line you get that it is looking for a makefile

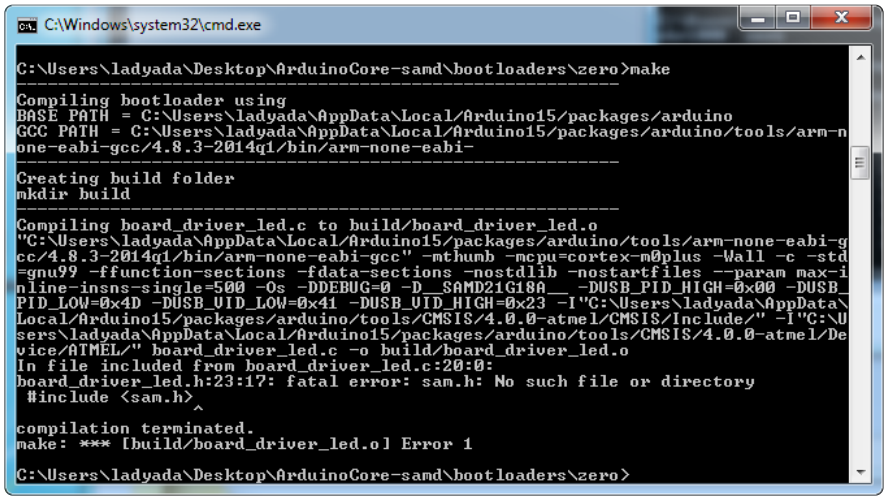


Change into the **bootloaders/zero** directory

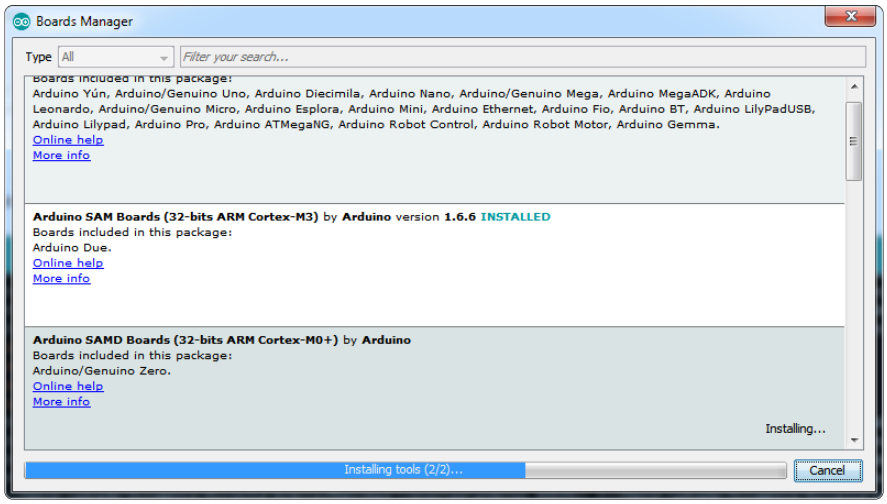


If you try to **make all** here and you get

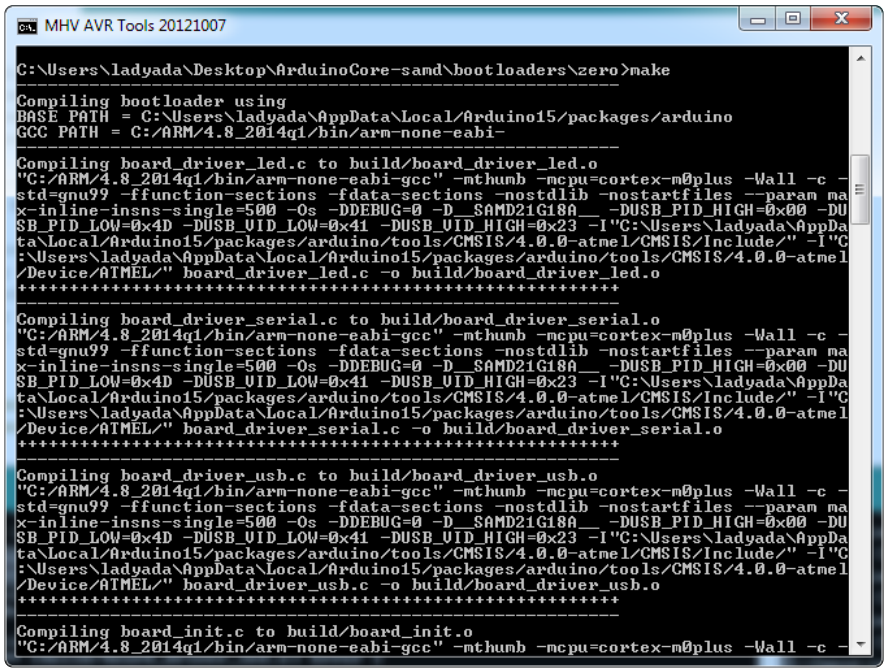
```
fatal error: sam.h: No such file or directory
```

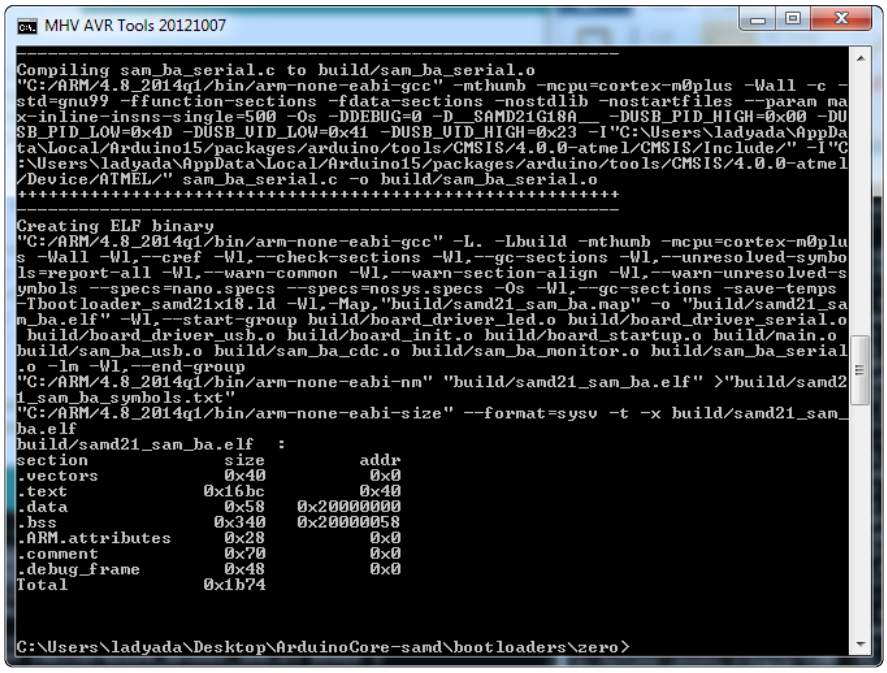


Make sure you've installed the Arduino SAM and SAMD packages through the board manager

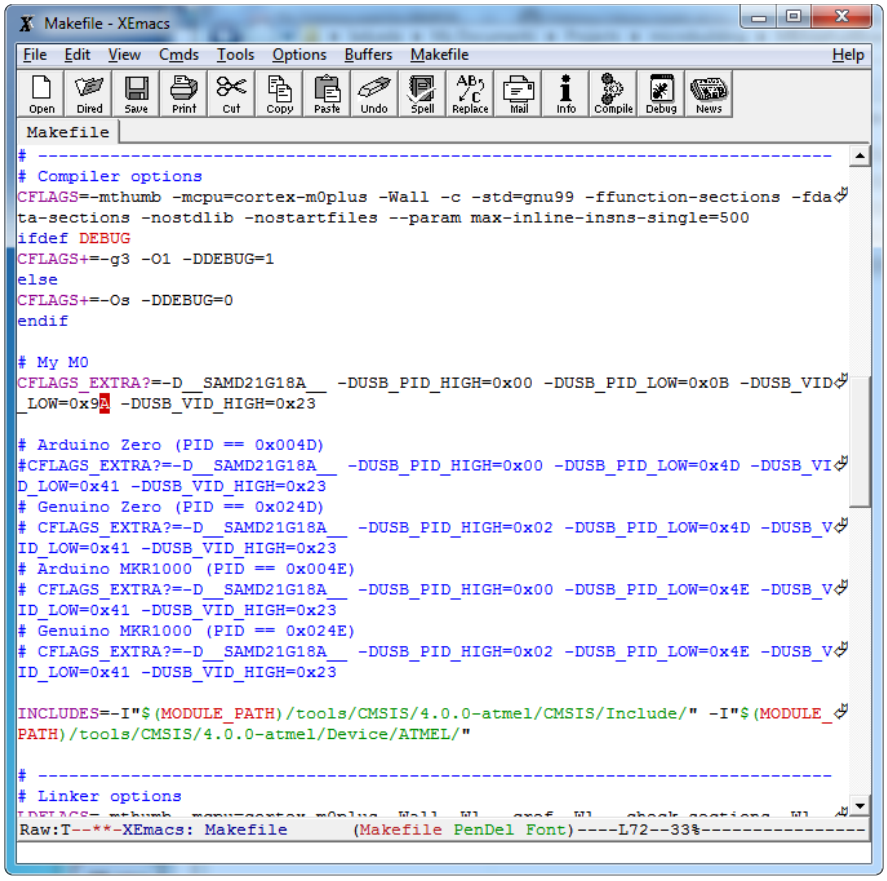


Try **make all** again for success!

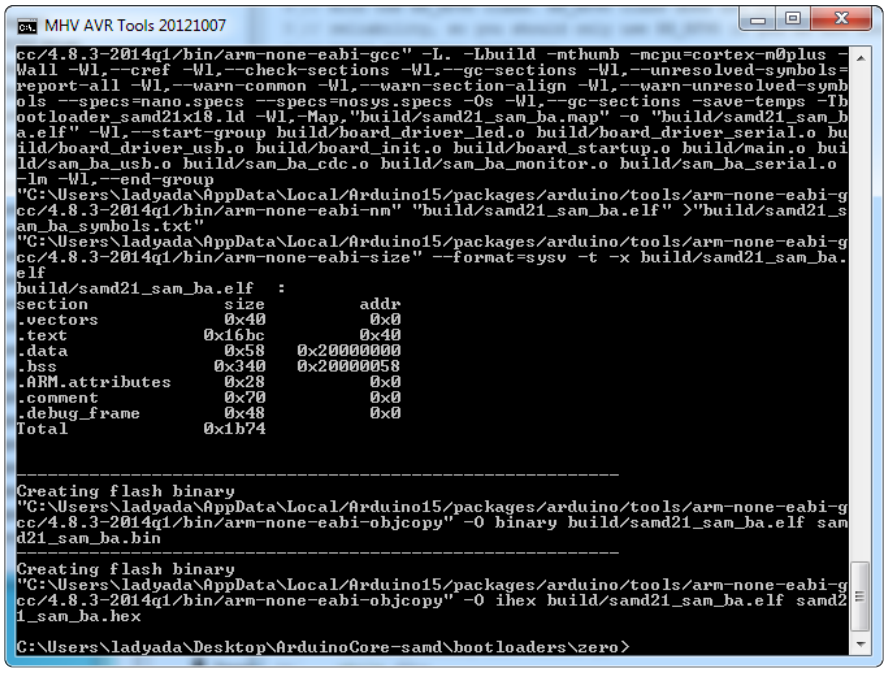




Before distributing, make sure to change the VID/PID in the makefile:



then run **make clean** and **make all** again to make fresh files



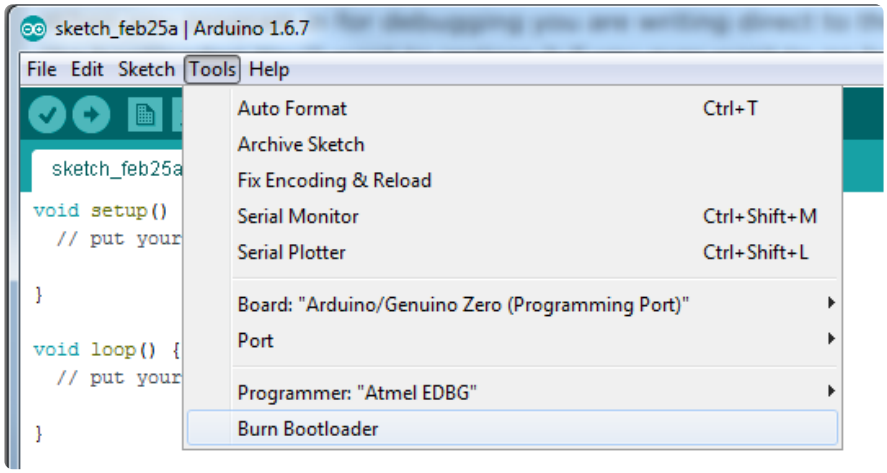
When you're done, you'll have two new files, **samd21_sam_ba.hex** and **samd21_sam_ba.bin**

Programming into an Arduino Zero w/EDBG

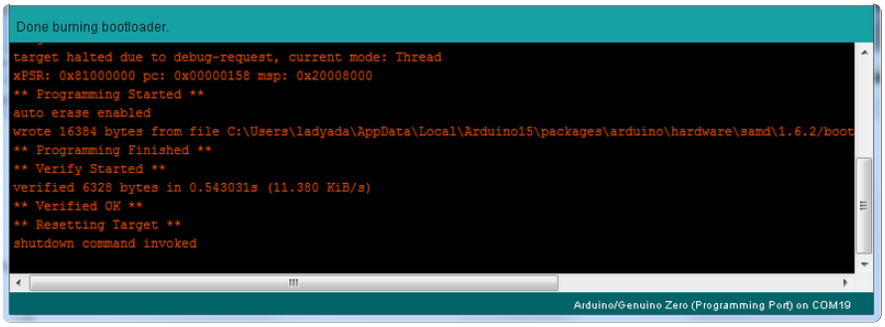
The easiest way to do this is just replace **samd21_sam_ba.bin** in **C:\Users\ladyada\AppData\Local\Arduino15\packages\arduino\hardware\samd\1.6.4\bootloaders\zero** (or where-ever the current version of your Arduino SAMD package is with the compiled **samd21_sam_ba.bin**

Launch the IDE, select **Arduino Zero (programming port)** from the **Tools->Board** menu, and **Atmel EDBG** as the **Tools->Programmer**

Then select **Burn Bootloader**



It only takes a few seconds to burn in the bootloader:



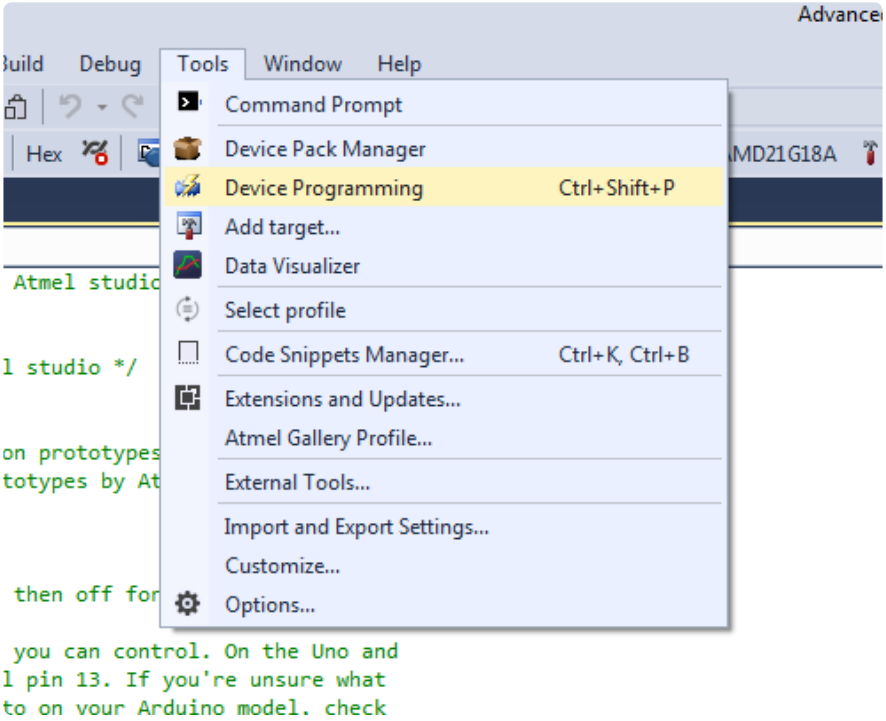
Feather M0 or Others

For this, you'll need to use an **Atmel Studio** or **adalink** setup, since you're using a J-Link or stlink.

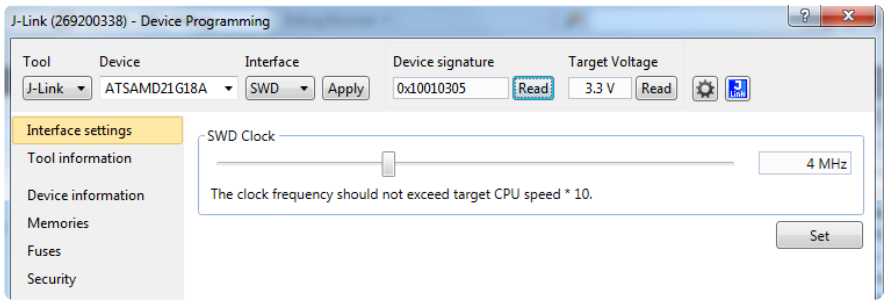
Atmel Studio

If you're using Atmel Studio, install it (https://adafru.it/mdm)

Plug in your Jlink and connect it to the SWD pins. OK now you have your debugger plugged in, its good to check that it works, select **Device Programming**



Under **Tool** make sure you can select **J-Link**

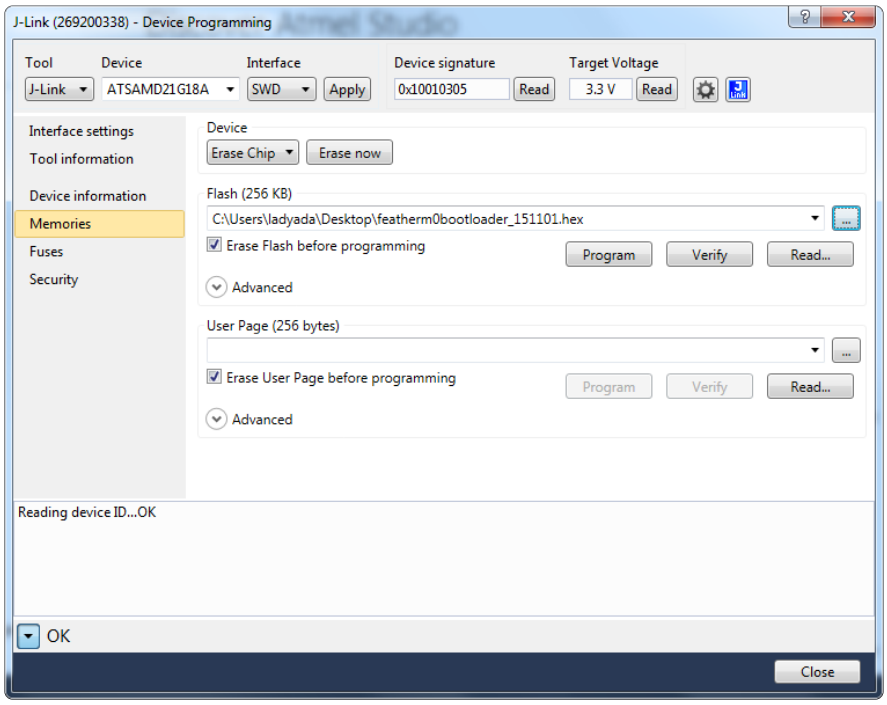


Select **ATSAMD21G18A** as the device, **SWD** as the interface and hit **Apply**

You can then **Read** the Device Signature. Make sure this all works before you continue!

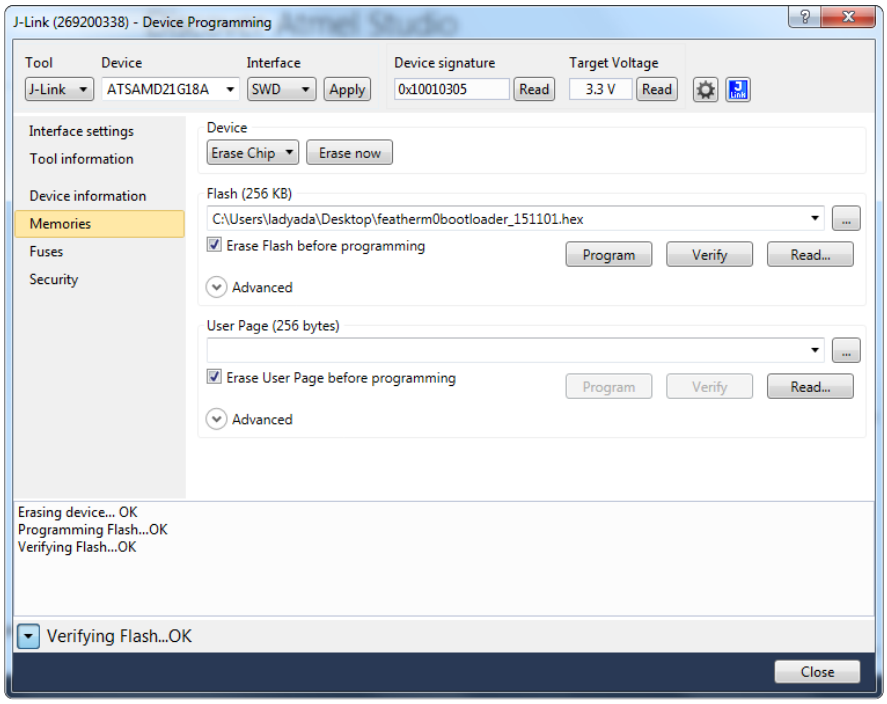
If you are asked to update the J-Link firmware, its OK to do so now.

Next click on **Memories** in the left hand side



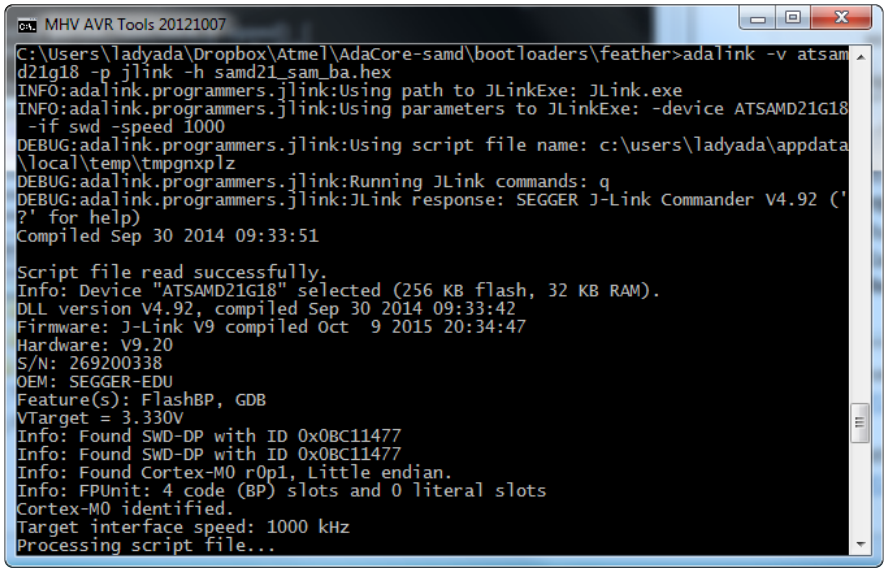
Next to the **Flash (256 KB)** section, click the triple-dots and select the **hex** bootloader file generated from the previous **make all**

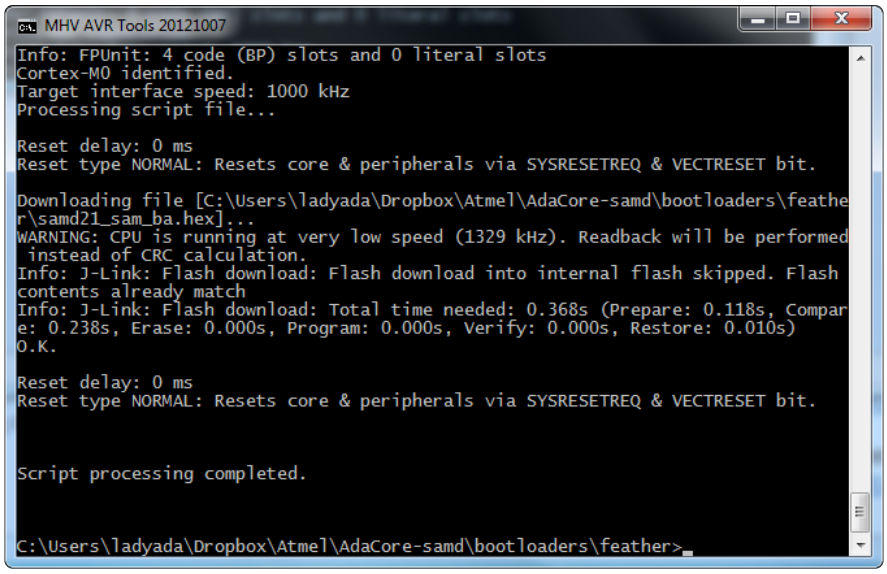
Then click **Program** to program it in



If you don't have Windows you can install **adalink** (https://adafru.it/fPq)

Then run **adalink -v atsamd21g181 -p jlink -h samd21_sam_ba.hex**





You can also check out https://learn.adafruit.com/programming-microcontrollers-using-openocd-on-raspberry-pi (https://adafru.it/mdn)if you want hints on using OpenOCD directly!