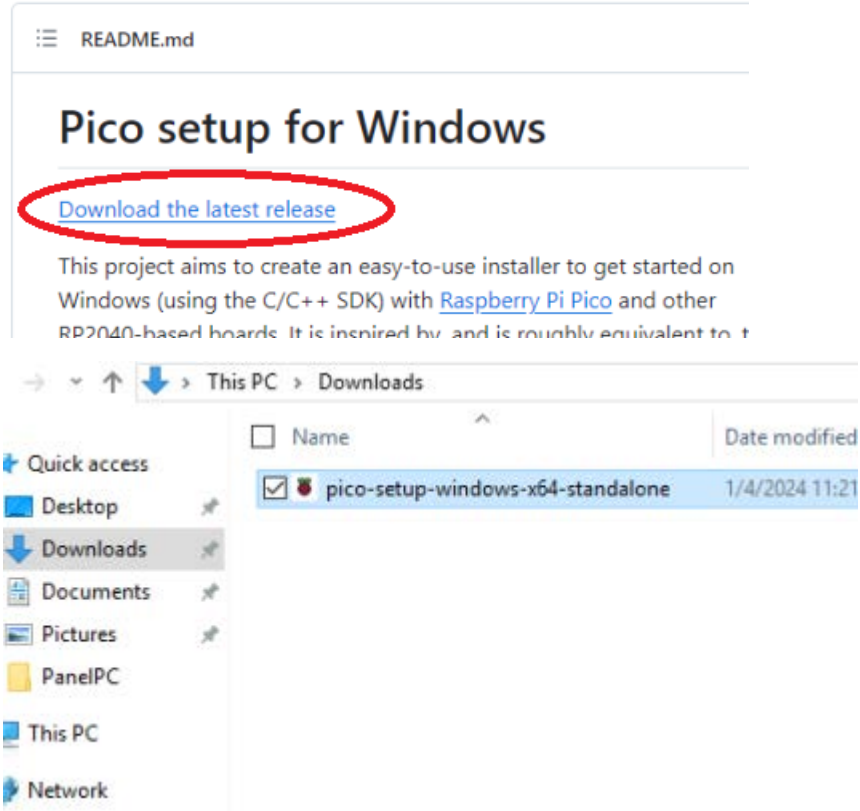
Idiot's guide to compiling grblHAL for RP2040 in Windows.

By Chuck Staton

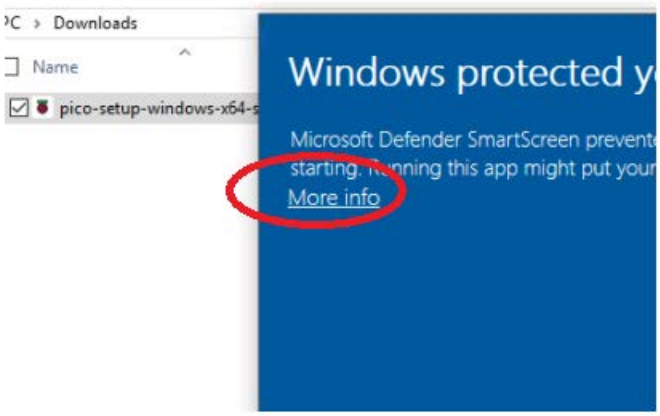
1/5/2024

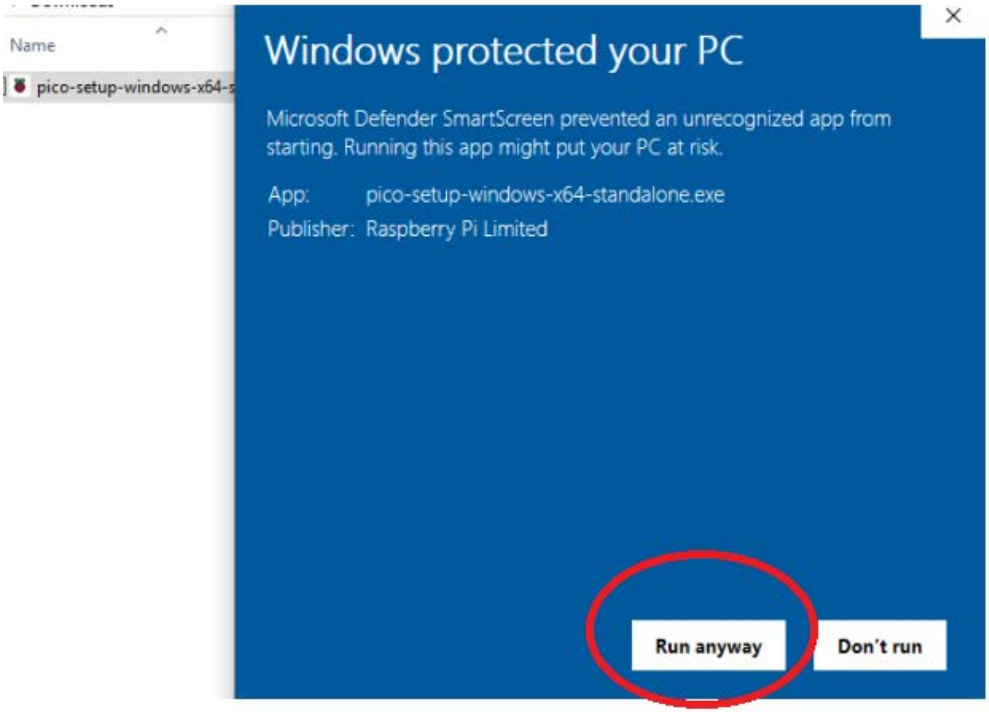
Download Pico SDK:

https://github.com/raspberrypi/pico-setup-windows?tab=readme-ov-file



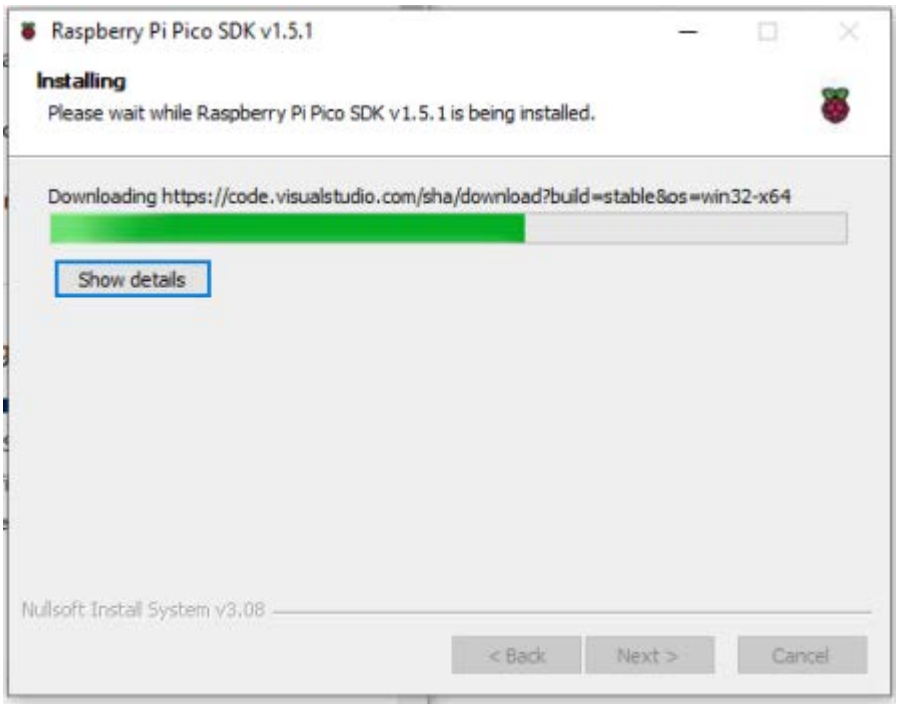
Click "more info" and "run anyway"

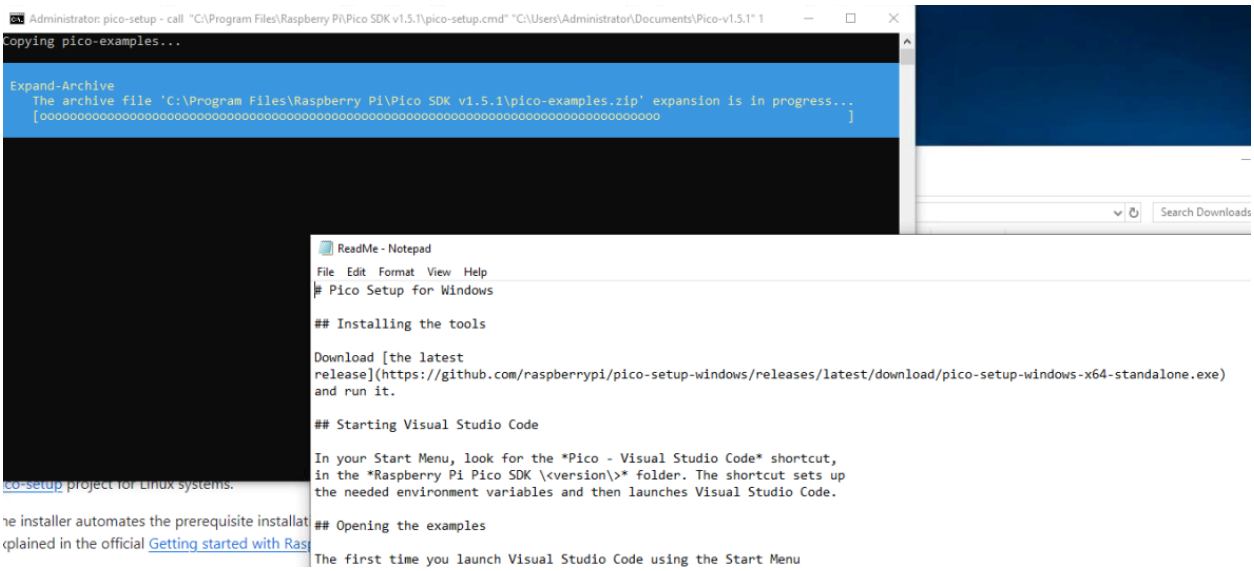




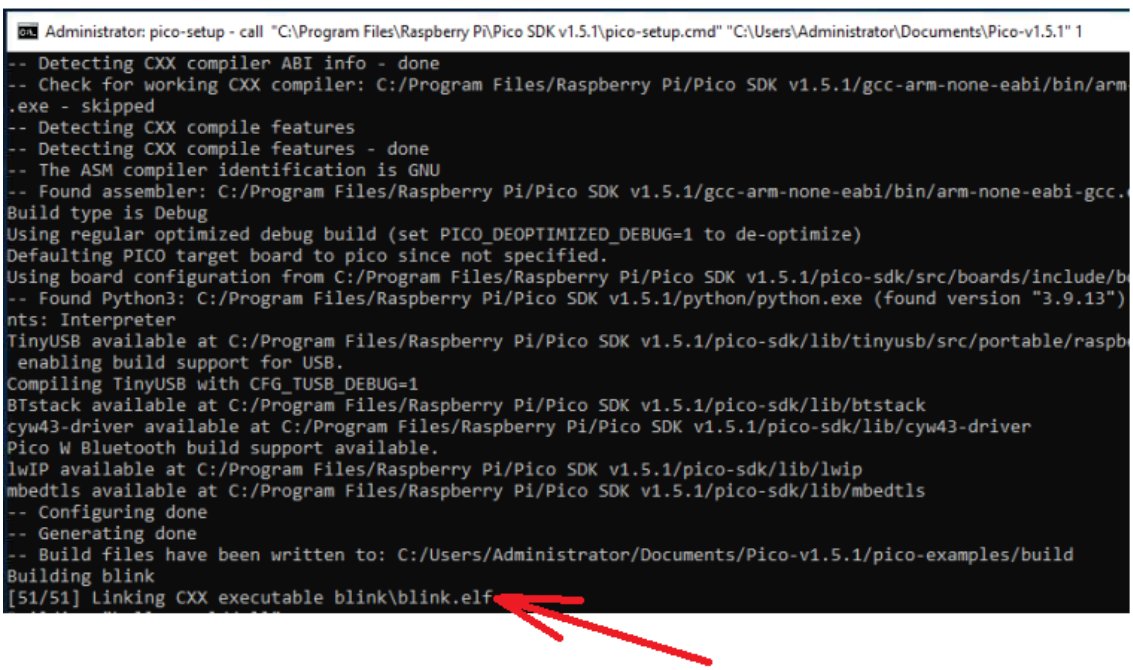
Click next/ok/install/yes/ok/next/whatever/finish (just click through the install)

This will take a while because it's downloading all the stuff you need including VS Code, Python, etc.



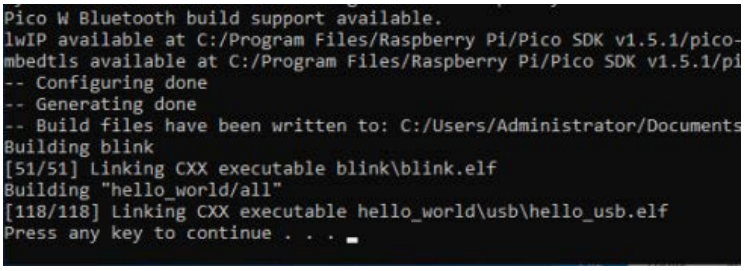


If it sits here forever saying [0/51], it's waiting for you to press ENTER for some strange reason:



Just hit ENTER and it will finish:

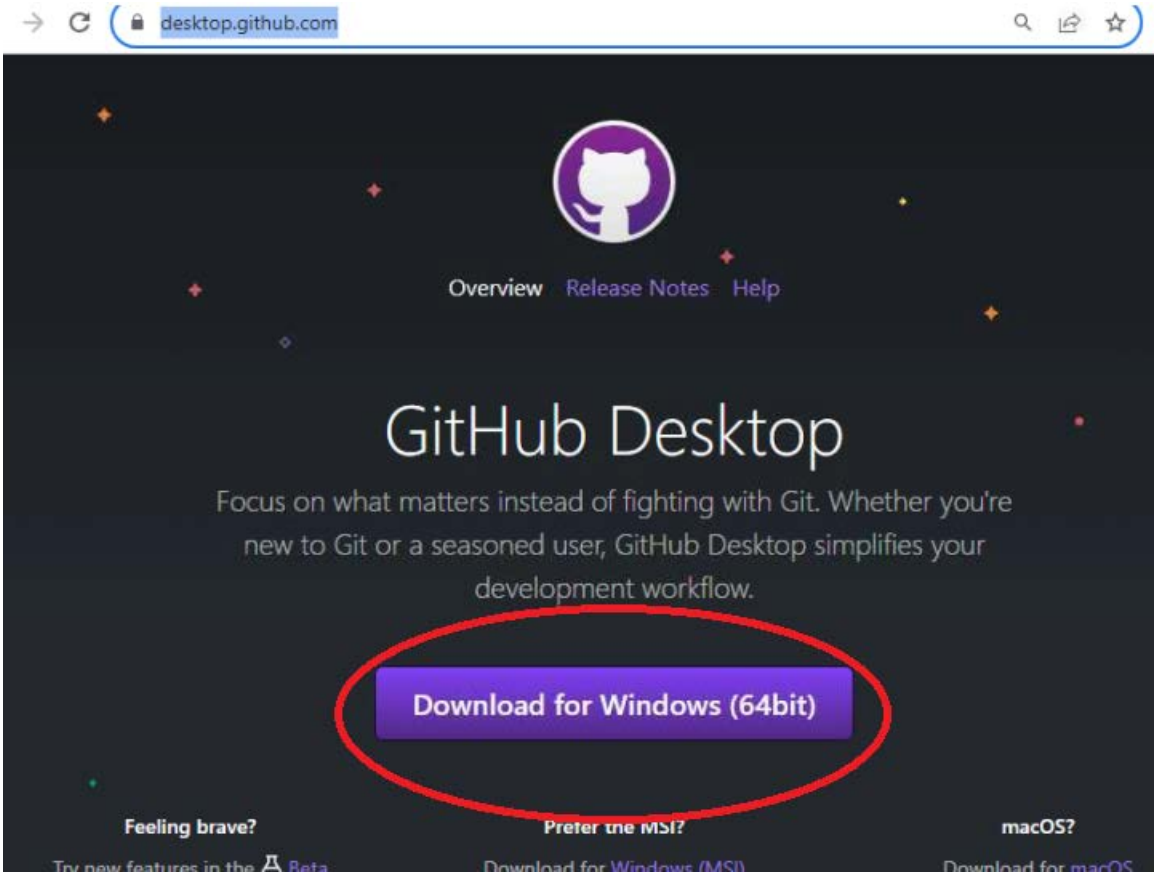
Then hit ENTER again when it gets to the end:

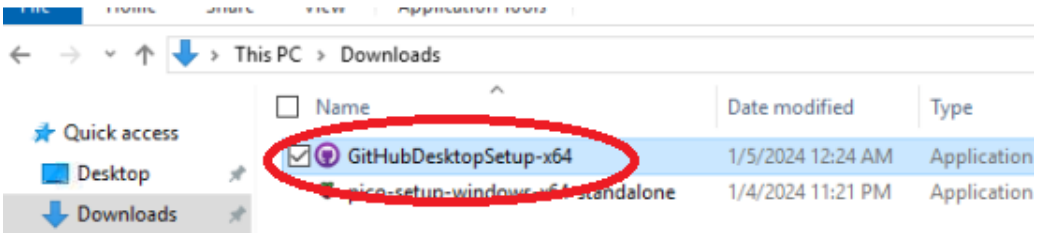


You can close the Pico-setup command window.

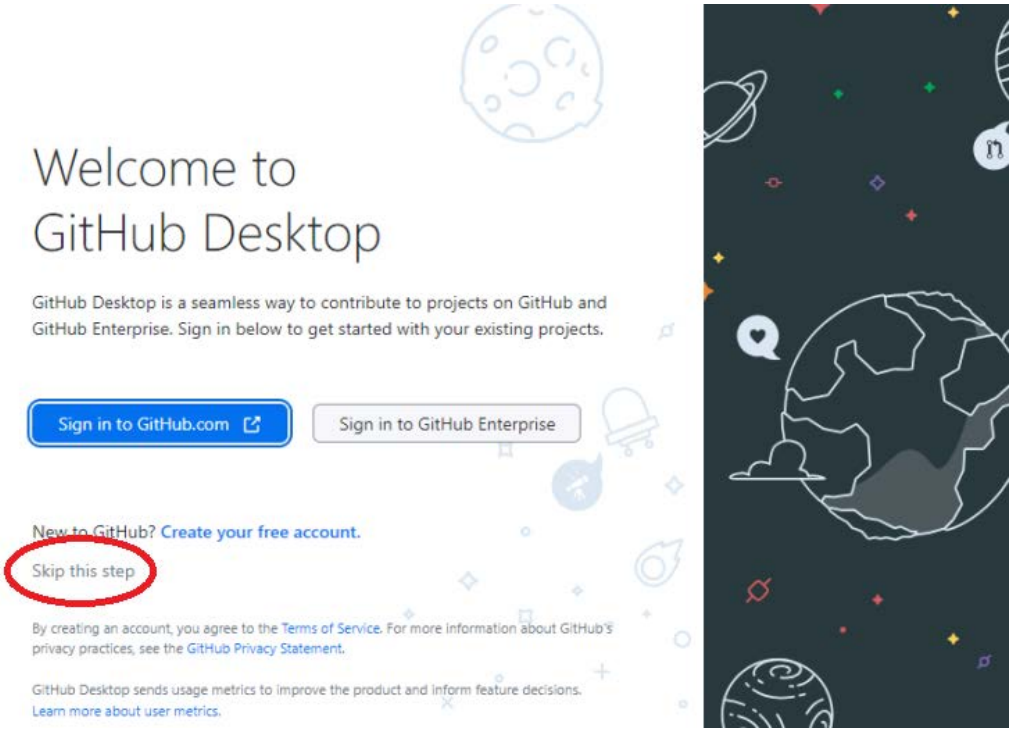
Download and install Github Desktop

https://desktop.github.com/

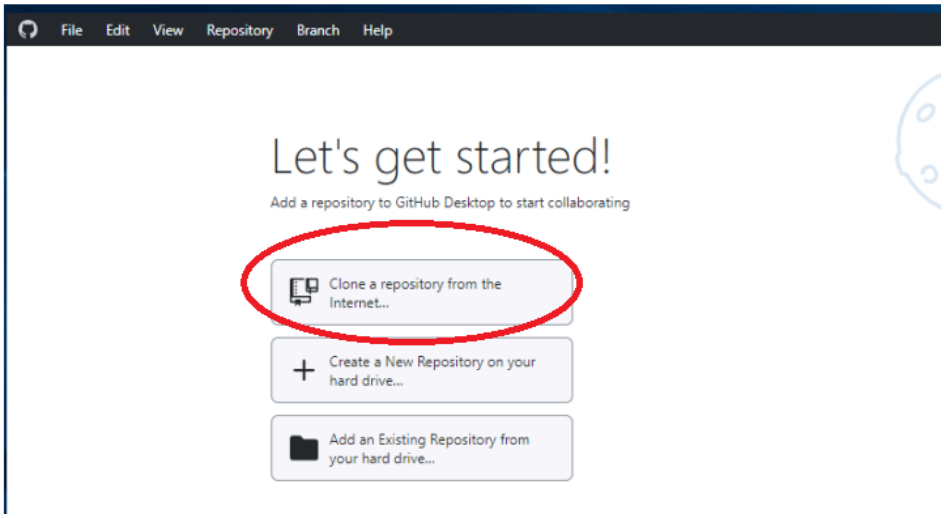




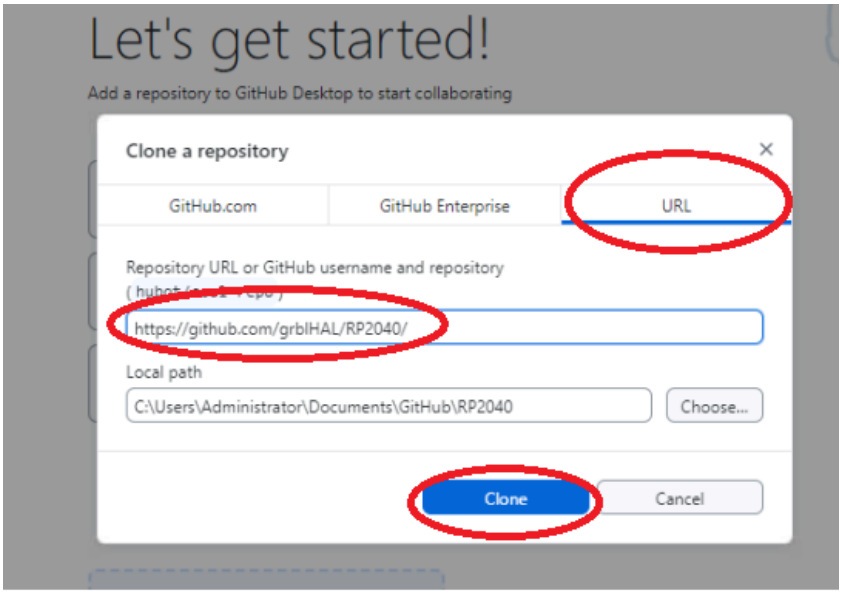
Don't bother creating an account if you don't already have one, and don't plan on committing anything



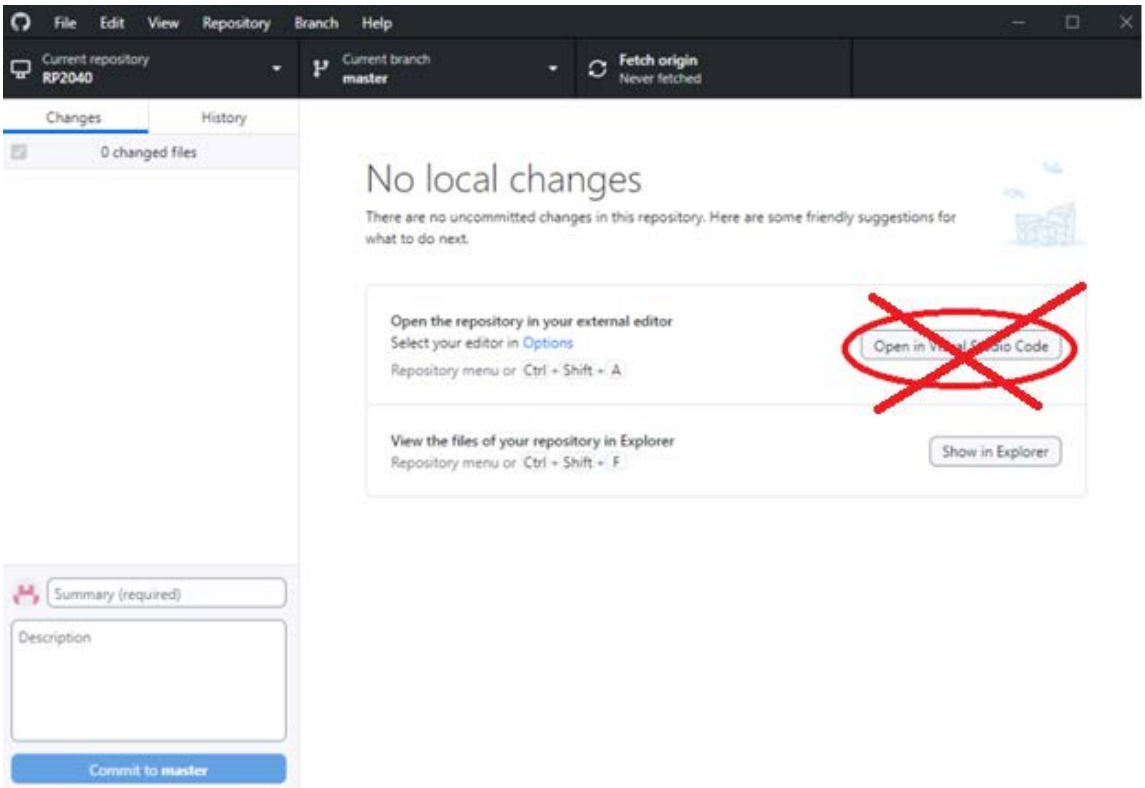
Click Clone:



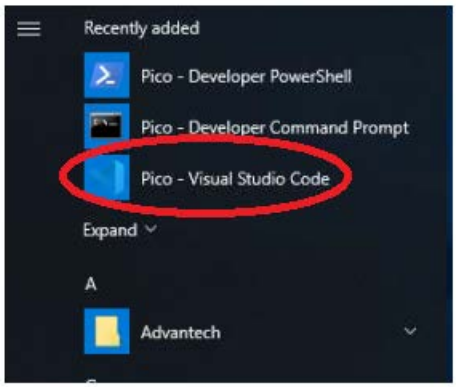
Click URL, then enter https://github.com/grblHAL/RP2040/, then click “Clone”



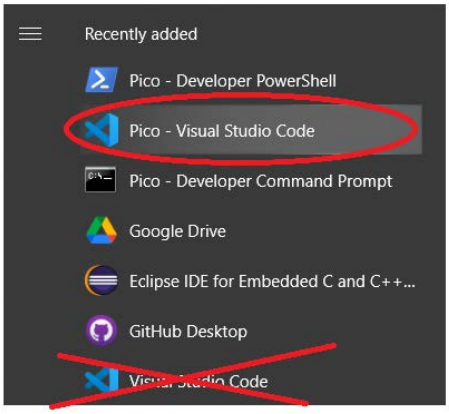
Once it's done, **<u>DO NOT</u>** click "Open in VS Code." For some reason this causes errors later on.



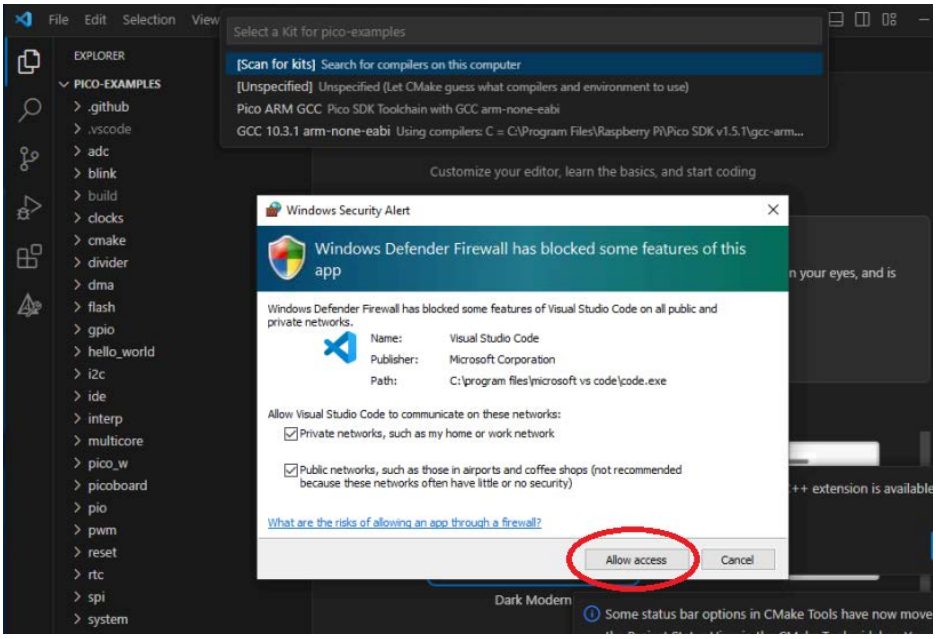
Close Github and open VS Code from the start menu



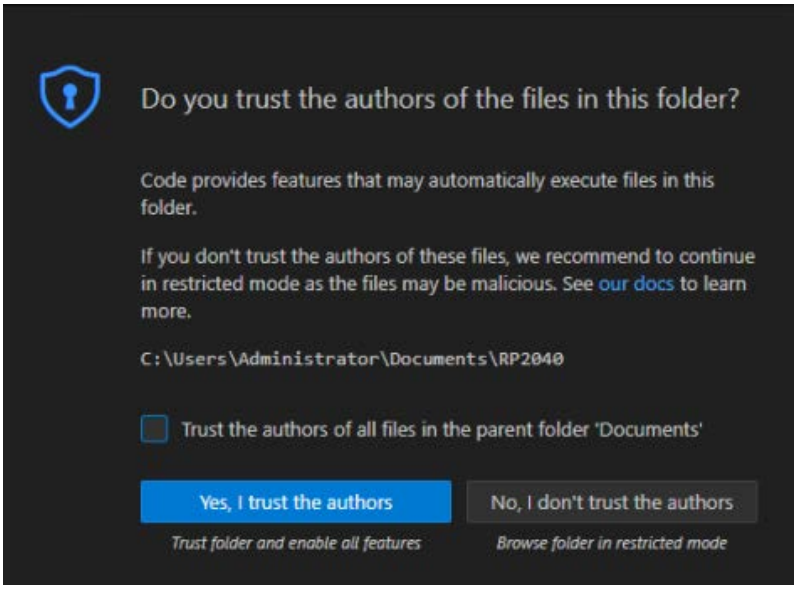
If you already had VS Code installed, make sure you open the new one that says "**Pico** – VS Code"



If this comes up, tick all the boxes and click Allow

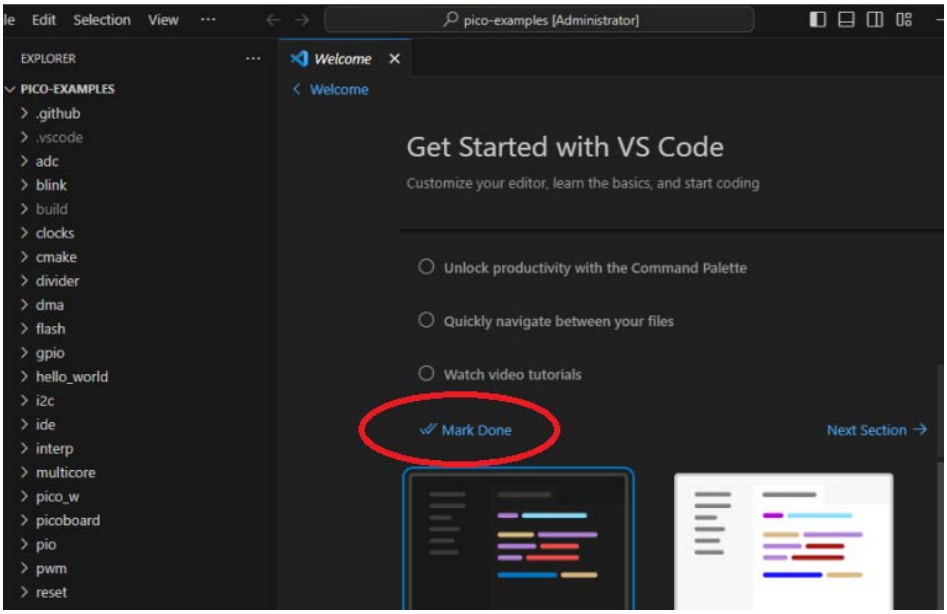


If this comes up, click YES:

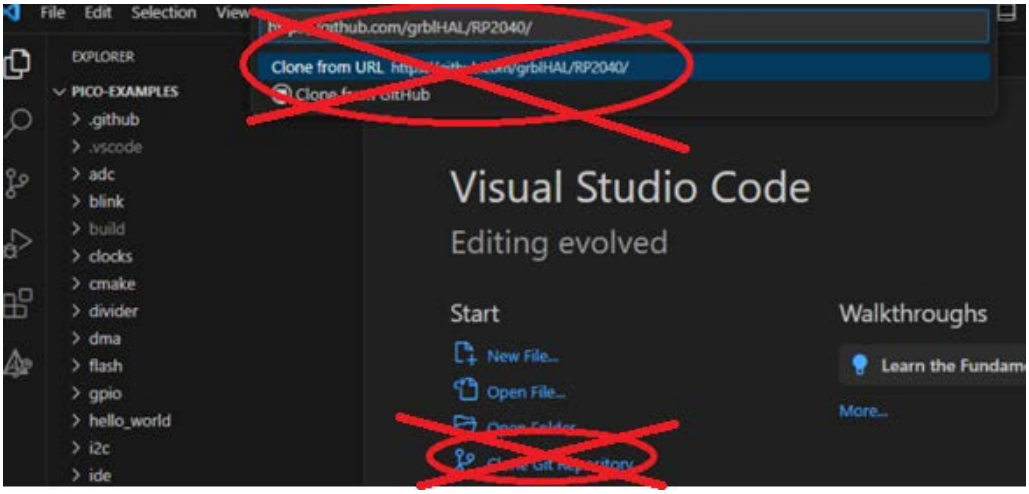


Click “X” on all the notifications in the bottom right. Don’t bother reading them, just close them.

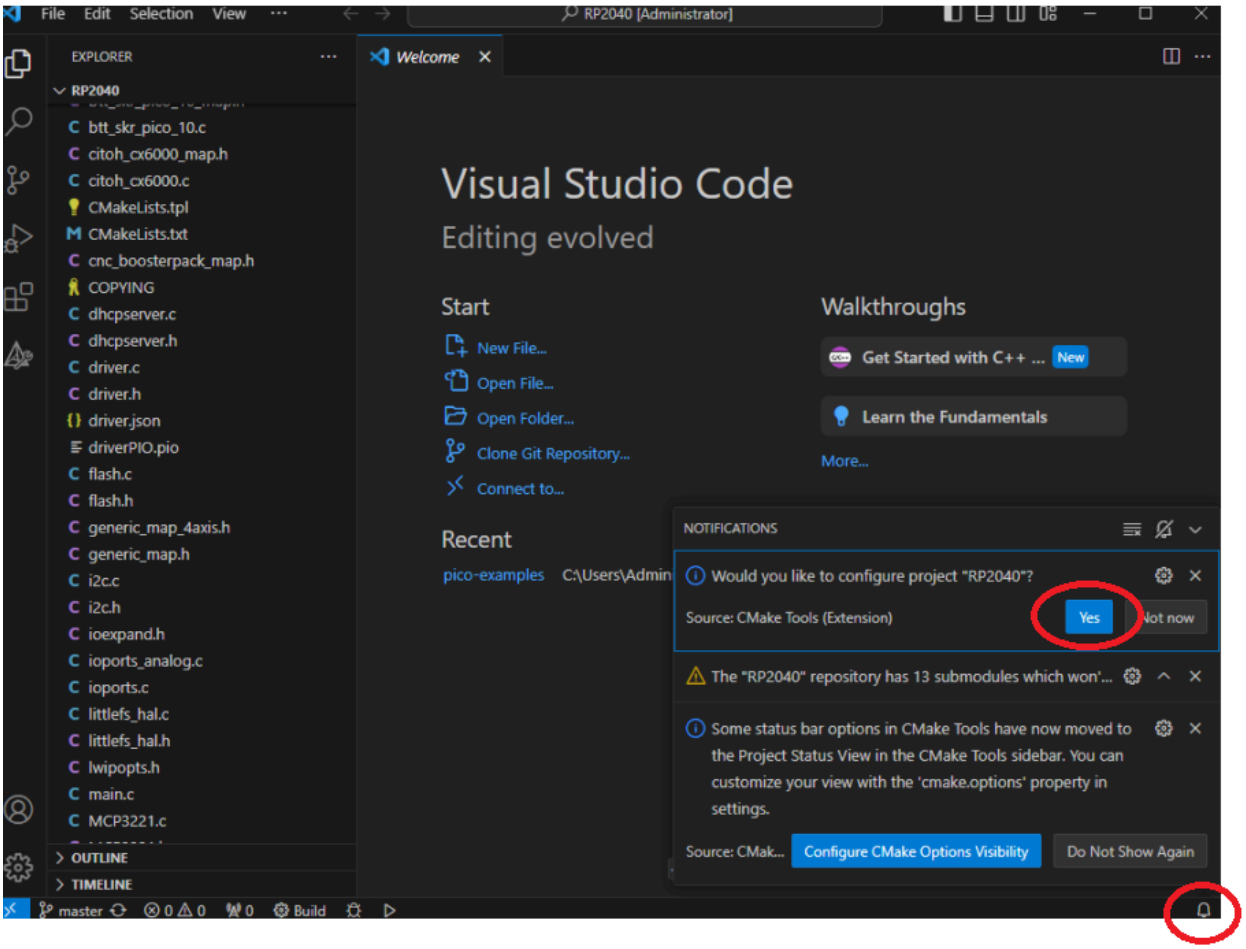
Skip all the “welcome to VS Code” nonsense by clicking “Mark Done”:



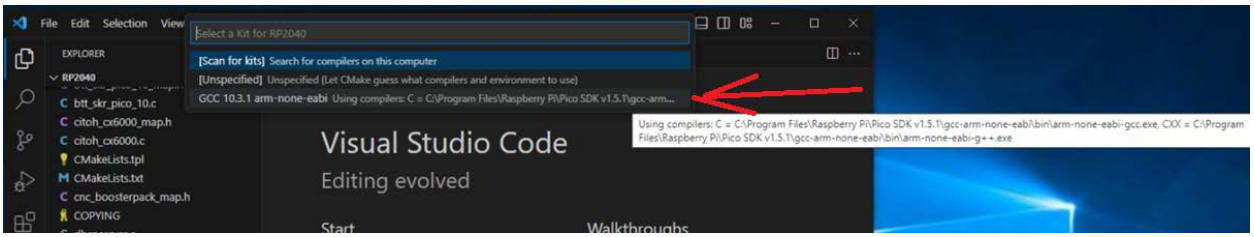
Note: Trying to clone the repository from inside VS Code does **NOT** work. Just use Github desktop as shown earlier..



After you open the cloned repo, there will be a message in the bottom right asking if you want to configure the project, click Yes. If you don't respond to the message in time, it will disappear, but you can get it back by clicking the bell icon in the bottom right.

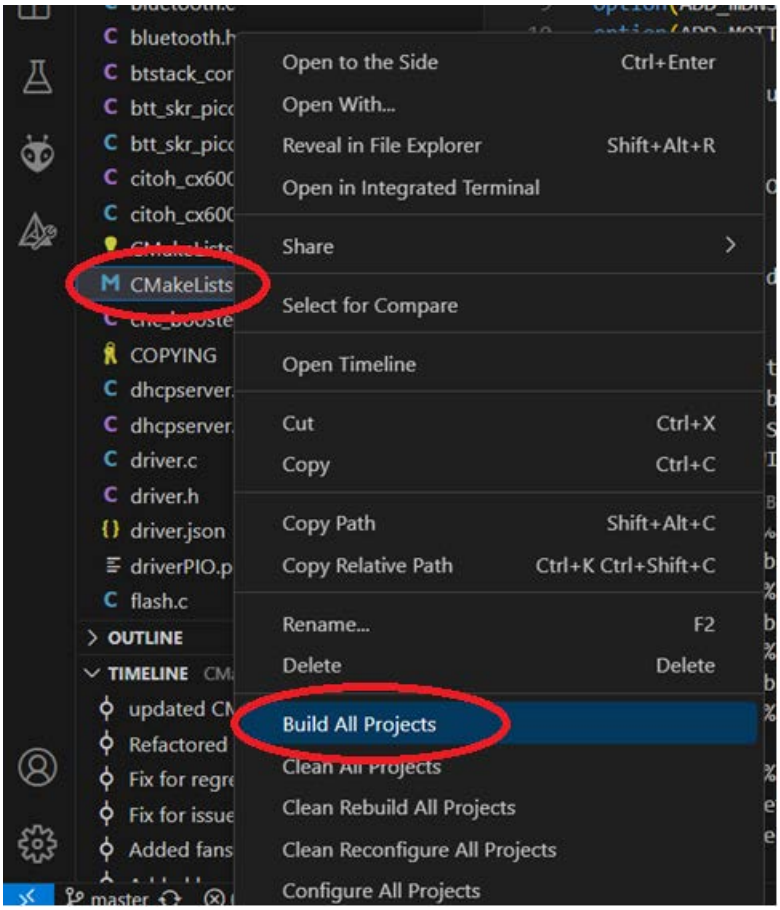


Once you click YES, there will be a list of option in the top bar. Choose the bottom one (GCC 10.3.1 arm-…)



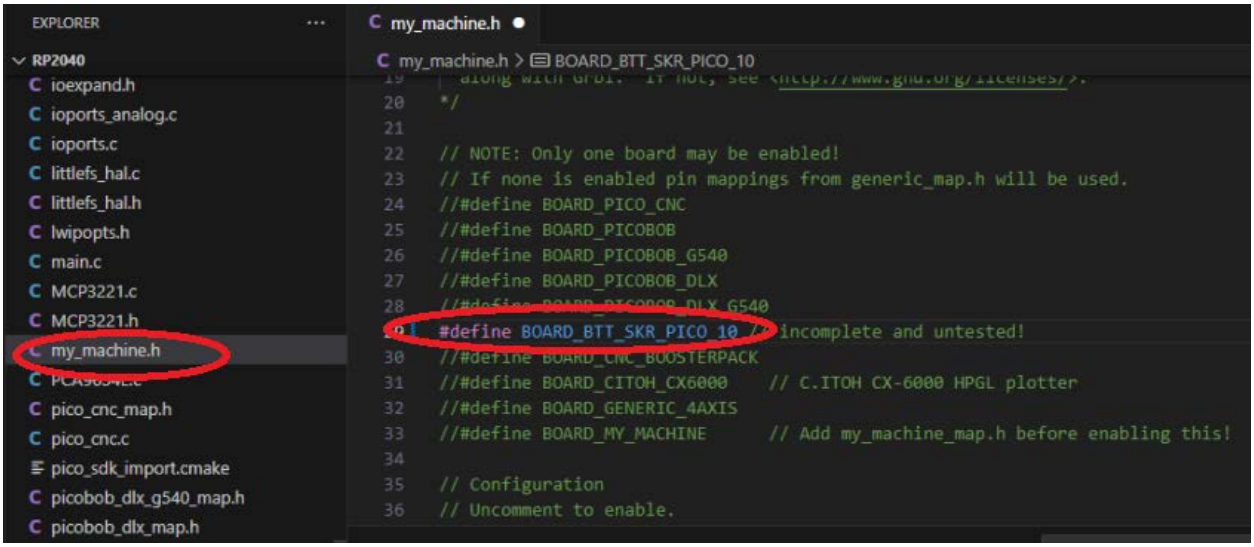
It will run some processes and spit out some text in the bottom of the screen. Check to make sure nothing in that box smells like an error (anything finishing with an exit code not equal to 0, is smelly).

Select CMakeLists.txt, right-click it, and click build all



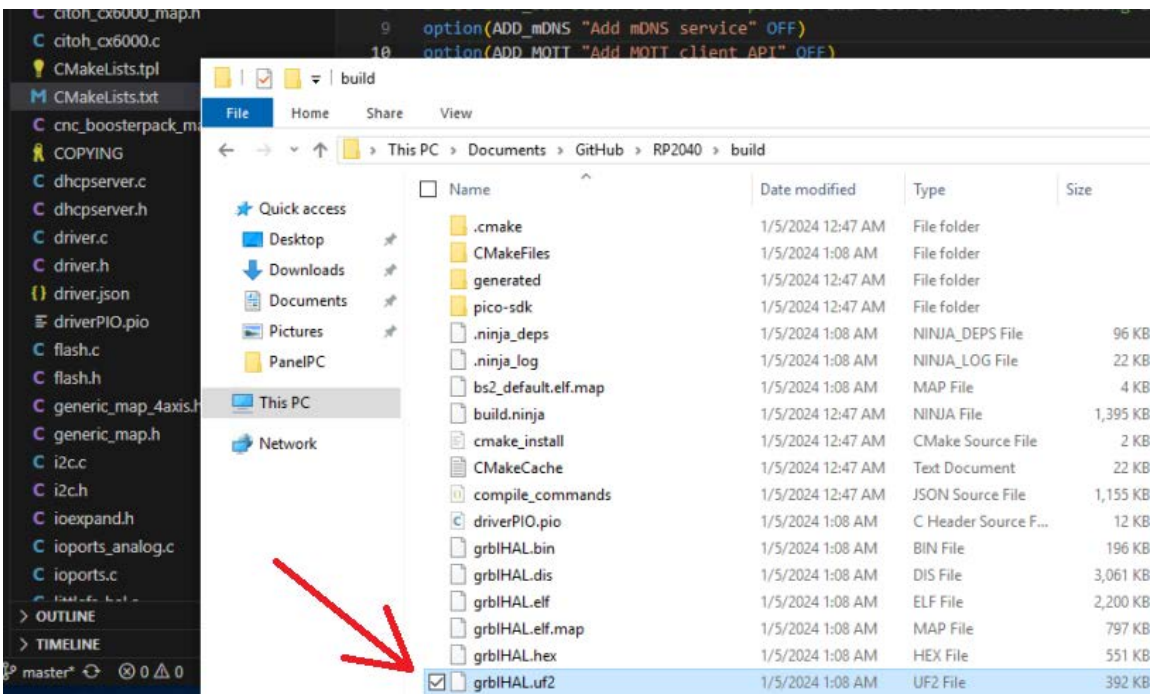
It will do a bunch of thinking, generate a bunch of monologue, and create a new folder ("build") in the project directory, but there won't be any firmware in it because we didn't select a board. This was just a test.

If there are no errors, you can select a board by opening my_machine.h and uncommenting the board you're using (removing the "//" in front of it).



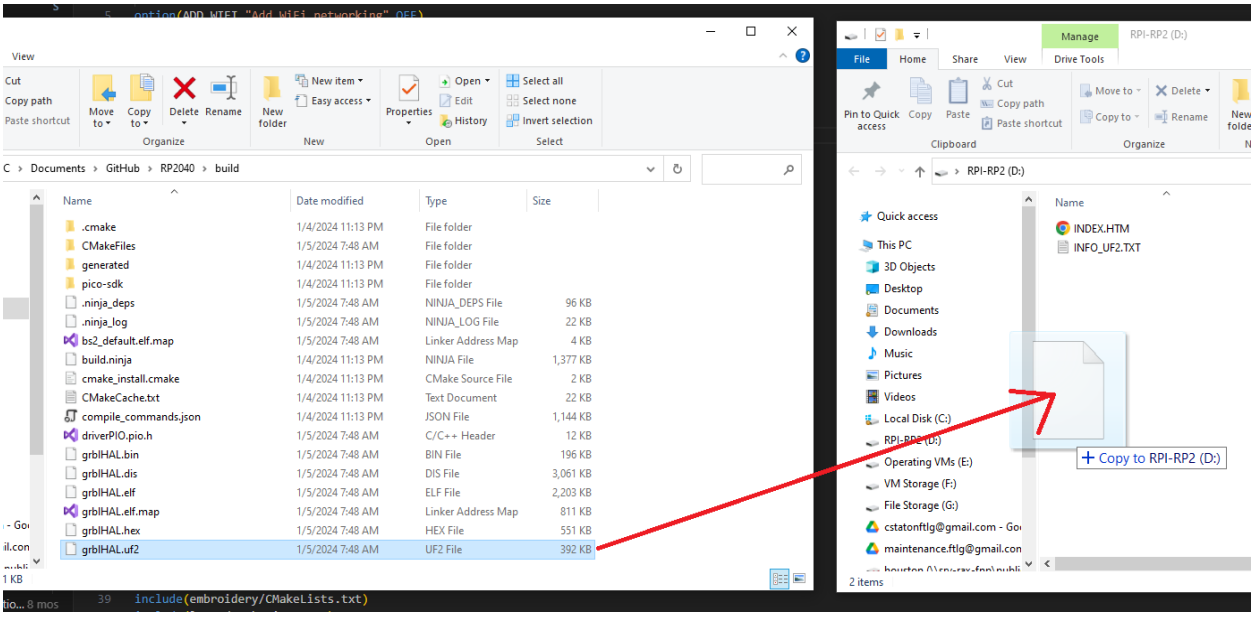
Now, again select CMakeLists.txt, right-click it, and click build all. Now you should have some usable firmware in the Build folder.

Go into the new folder (build) and the grblHAL.uf2 file is the firmware you just created.



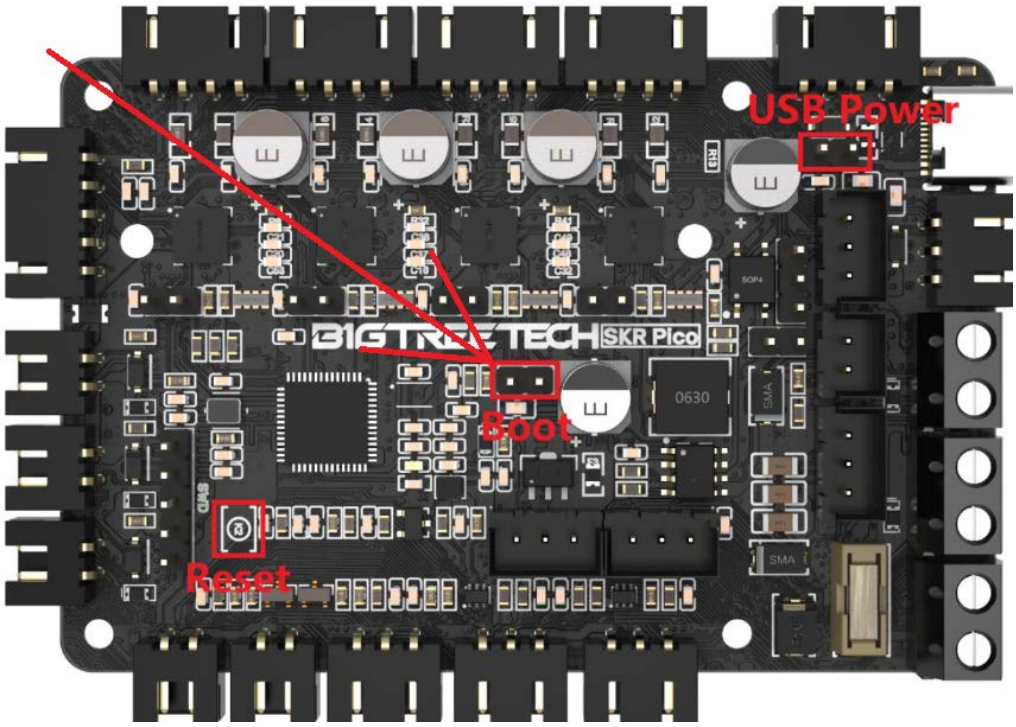
If you're happy with it, drag and drop the grblHAL.uf2 file into the board. When you first plug in a RP2040 board, it comes up as a USB mass storage device.

You just drag grblHAL.uf2 out of the GitHub\RP2040\build folder and drop it into the RPI-RP2 device.



The board will reboot and disappear. If it disappears, that means it worked, because it now has firmware and has booted into its normal mode.

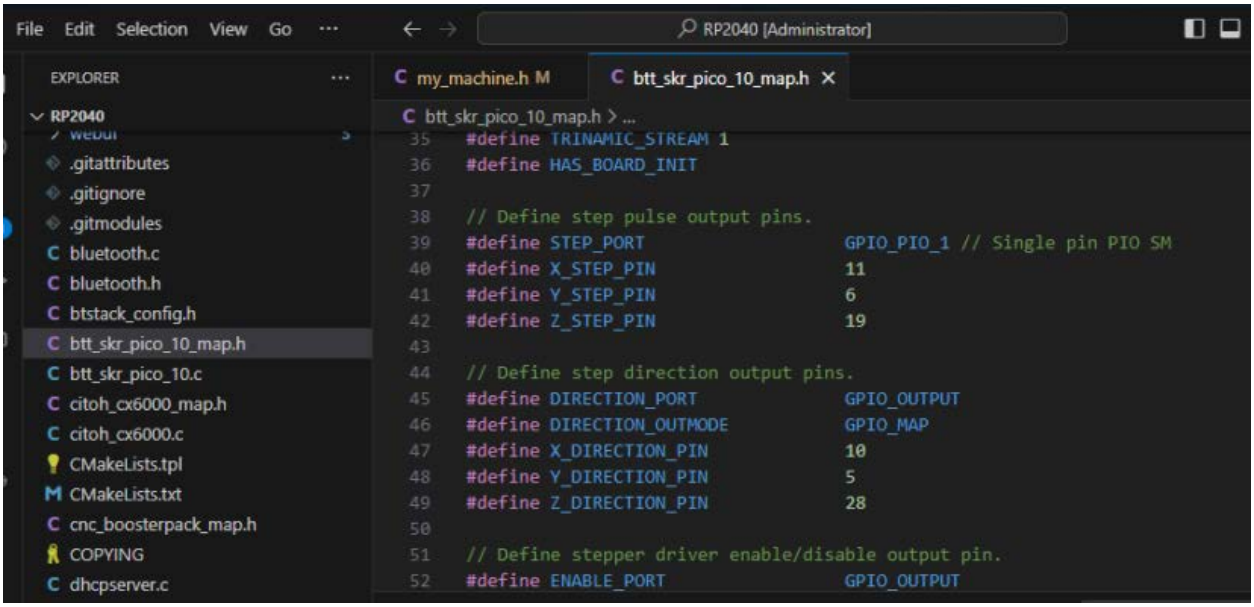
On some boards (like mine, the BTT SKR Pico board) there is a jumper that you must install in order for this mass storage device mode to be turned on. And you must remove the jumper once firmware has been installed in it, or it will keep booting into mass storage mode.



A normal Raspberry Pi Pico board does not have this jumper, so it comes up in Mass Storage mode only the first time. If you want to reinstall/update firmware onto it, you must hold down the “bootsel” button while you plug it in.

If you want to make changes to the firmware, make them in…

- The pin map for your board (in my case GitHub\RP2040\btt_skr_pico_10_map.h):



  This is where you change which pins are assigned to which functions. In my case, I want to utilize some of the unmapped pins as AUX relay outputs repurpose some of the mapped pins for other uses
- GitHub\RP2040\my_machine.h
  - This is where you select the board you are using and enable optional functions.
- GitHub\RP2040\grbl\config.h
  - More advanced settings here, you probably don’t need to change anything.

…but leave everything else alone unless you know what you’re doing. If you needed this guide, you should probably stick to those 3 files. Make your changes, and then select CMakeLists.txt, right-click it, and click build all, then **rename the new grblHAL.uf2 file** and drag/drop the new file onto the board. The firmware file should always have a different name each time you send it to the board.