Troubleshooting

As with all things programming, things have a habit of not working on the first try, and that's OK! Trying to figure out what is going wrong with your code might be a little tricky with MicroPython, but it can be done.

Cannot Upload Program

If you cannot upload your program and get an error such as ampy.pyboard.PyboardError: failed to access COM7 from ampy, it means that another program is using the serial connection to your ESP32.

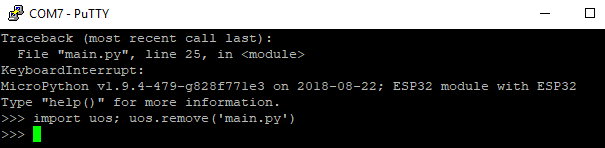
Take a look at all of your open programs and close or disconnect anything that has an open serial connection to your ESP32.

If you are on Windows and your port number is higher than 10, ampy may give you an error message like so: ampy.pyboard.PyboardError: failed to access \\.\COM24. There is a known issue with Python's recognition of Windows ports above 10, but ampy does not take this into account, so we must format our port explicity for Python.

Try formatting your COM port as you see it here (replacing <PORT> with your port number/location): \\.\<PORT>

If, on the other hand, ampy gives you an error message like ampy.pyboard.PyboardError: could not enter raw repl, it means that the ESP32 is running a program and not in REPL mode. If you were following this guide exactly, you should be able to press button 0 on the ESP32 Thing to exit your program to REPL. If you forget to add some "Drop to REPL" code in your program, you will need to manually stop your code and remove *main.py*:

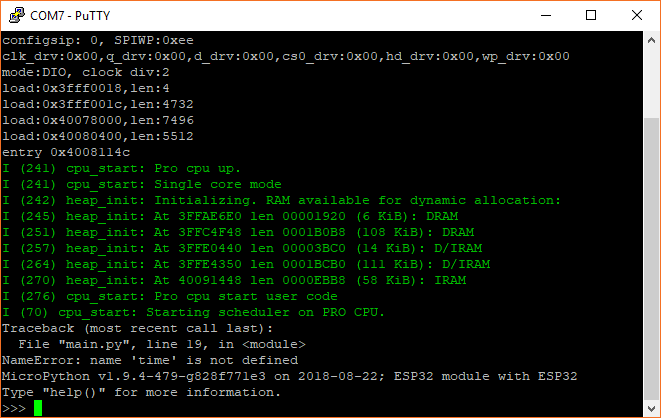
1. Open a serial terminal program and connect to your ESP32.
2. In your serial terminal, press ctrl + c to send the *SIGINT* interrupt signal to the program running on the ESP32. You should see a REPL prompt (>>>) appear.
3. In REPL, enter the following commands: import uos; uos.remove('main.py'). This will import the uos module, which allows us to manipulate files and then deletes the *main.py* file, which will prevent our program from running in the future.
4. Close your serial terminal program and try uploading your *main.py* again.



Debugging

If you were able to upload your program but nothing runs, you could have a syntax error in your code or a runtime error.

The first step is to use your serial terminal program to connect to your ESP32. Press the reset button (labeled *RST* on the board). If there are syntax errors in your code, you should see them printed out to the serial terminal.

[](https://cdn.sparkfun.com/assets/learn_tutorials/6/5/9/screen_12.png)

*In this example, you can see that we wrote "time" instead of "utime" on line 19 of our code. If you see an error message after restarting your ESP32, always read it carefully!*

If you do not see any syntax errors, you may have a runtime error (e.g. your code is doing something that you did not intend for it to do). There are many ways to debug runtime errors, but one of the most common methods is to add print() statements in your code to see what is happening at that moment in your program (or to check if your program is getting to that point at all).

[](https://cdn.sparkfun.com/assets/learn_tutorials/6/5/9/screen_13.png)

*Here, we added a print(i) line to our for loop in the blink.py example to ensure that i counts up to 1023 before resetting back to 0*