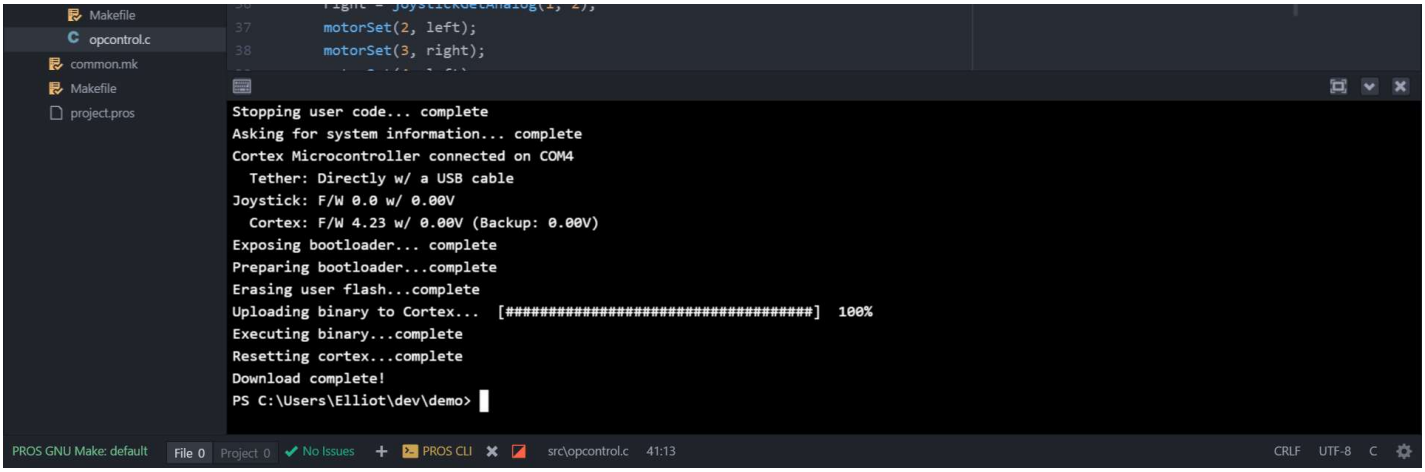
Uploading PROS Projects

The PROS uploading utility allows you to flash compiled binaries to your VEX microcontroller. There are a variety of different ways to flash code:

- USB A-A (Windows only)
- Programming module w/USB between Joystick and Microcontroller
- Programming module w/VEXnet between Joystick and Microcontroller

Within Atom, click the 'Upload to Cortex' button in Atom. This process will upload the binary the compilation process created (`bin/output.bin`) to the Cortex and begin running it.



Using just the terminal, this can be done by navigating to the directory (or subdirectory) of the PROS project and invoking `pros upload` . You can get all of the command options by running `pros upload --help` . Some additional commands you may find helpful is `pros mu` , which runs the default Make target (forks and calls `make`) and then uploads your code, and `pros mut` , which does the same but also opens the PROS terminal after uploading.

Having issues flashing?

⚠ PROS CLI 2.5.0 Available

We fixed a number of issues flashing code onto the VEX Cortex. Make sure you've updated PROS to at least 2.5.0.

Before trying these solutions, make sure you have built your project! If there's nothing to upload, the flasher will throw an error!

Use USB

First, try putting the fewest amount of electronics between the Cortex and computer. If you're on Windows, try flashing using only a USB A-A cable. If you're on a Unix-like OS, flash using a programming module with a USB cable between the Joystick and microcontroller. Sometimes commands to upload your code are dropped when flashing wirelessly, especially when the distance between the joystick and microcontroller is large.

Don't poll

Next, try using the `--no-poll` option. This limits communication between the Cortex and your computer to just what's vitally important to flashing. Polling allows us to determine what connection type you're using and also allows you to check what firmware version you're running. Sometimes when printing a lot of data to the console, responses from the Cortex about system information gets intertwined with your data, preventing us from parsing it. Using `--no-poll` may allow you to bypass this issue.

Check for updates

There may be updates for the CLI which improves flasher reliability. If you keep Atom up to date, we'll notify you about updates. Otherwise, check Releases for the latest version.

Open an issue on GitHub and include debug output

You've tried everything you can and flashing still isn't working. Use the `-vd` option to print out all of the microcontroller's responses to the flasher's commands. Please copy the output of the program output and open an issue or shoot us an email at pros_development@cs.purdue.edu .