STELLA programming instructions

1. Plug the instrument into your computer. It should show up as an external drive named CIRCUITPY

2. On the underside of the display, in the finger notch in the housing on the left side, there's a reset button. Click that twice. It's a bit timing sensitive, so you might have to try this more than once. When you are successful, the CIRCUITPY drive should eject, and be replaced by FEATHERBOOT.

3. In the repository, there is a folder called current_uf2, and in that folder there is a file with a long name, ending in 5.3.1.uf2. It's the only file in that folder. Drag and drop that file onto the FEATHERBOOT drive. It will eject itself, and return as CIRCUITPY. It will now be running the version of the Circuit Python interpreter that matches the library file versions.

4. In the repository, there is a file called lib. Copy that file, including the folder, onto the CIRCUITPY drive.

5. In the repository, there is a folder called STELLA_FS_instrument_code. Open that folder. There's an archive of older versions, ignore that. Copy the code.py file to the CIRCUITPY drive. If everything is working correctly, the instrument should boot up and start making measurements. If not, we will need to go through some debugging.

6. Go to the webpage codewith.mu and download the Mu Editor.

7. Install and run the Mu Editor (you'll need administrator privileges to install it.)

8. Click on the Serial button in Mu, to look at the output of the STELLA instrument. If you click in the serial dialogue panel, ctrl-c will cancel the program, stopping it from running, and ctrl-d will decancel, restarting the program. You'll see error messages in this panel that we will need for debugging.

9. If it's all working correctly, that's super! The clock still needs to be set, just once for as long as the clock battery lasts. It should be set to UTC, universal coordinated time, which you can get here: https://time.is/UTC
In the repository, in the folder titled test_codes, there is a folder titled real_time_clock. From that folder, copy code.py to the CIRCUITPY drive. It will run the real time clock read and set program. In the Mu editor, click the Load button to load the program. The first if False: statement contains a place to set the clock. Set it to True, change the date, time, and day in the t = time.struct…. line, and then run the program to set the clock. Change it back to False, and re-run the program to confirm the time is set properly.

10. Then copy the instrument code to the instrument again, exactly as you did in Step 5.

At this point, it will either be working, or there will be some trouble in the wiring or the components. Look around in test_codes if you want to try debugging it yourself, or check with me and we can work through it together.