**Friday, November 3rd, 2023**

I worked on getting the motors controlled programmatically. I identified the motor controller on the car as an Enertion Focbox 1.2. This motor controller falls under the larger class of “VESC” type motor controllers which incorporate odometry by measuring fluctuations in the eddy currents from the brushless motor. The issue with this is that it has an older firmware version not recognized with newer versions of the VESC firmware uploader. For now, I will be keeping it on the old firmware. However, I did search extensively for a newer firmware I could flash onto it but did not find a trustworthy source. Thus, if this does not end up working it might be necessary to buy a new motor controller.

I then downloaded and installed the PyVESC library on my computer (<https://github.com/LiamBindle/PyVESC>) which provides a wrapper on the low-level serial communication which occurs over USB. The Universal nature of USB means I can control it either through my laptop or the Jetson board, but it’s easier to rapidly iterate on my laptop – the code will transfer to the Jetson board with minimal changes to account for the different operation system.

**Wednesday, November 8th, 2023**

I continued working on getting motor control up and working. I first filed two issues on GitHub asking for help:

* <https://github.com/vedderb/vesc_tool/issues/336>
* <https://github.com/LiamBindle/PyVESC/issues/31>

Then, I examined the source code of the PyVESC library to try and figure out why the motor controller wasn’t communicating. I found that the VESC was sending 3 bytes as its firmware version but the library expected 4. However, I was hesitant to change the library in the event that the firmware is really supposed to be a 4-byte long message and reducing it would offset the reading of all further messages by 1 byte. The serial protocol the VESC uses is quite opaque and hard to understand so I ultimately ended up not messing with it too much (I don’t want to brick the built-in microcontroller).

I found a tutorial for programming my particular model of the motor controller which I will try out on Friday (<https://jetsonhacks.com/2018/02/13/racecar-j-programming-the-electronic-speed-controller/>).

Once I’ve got motor control working I’ll assemble the car and it will be ready for reinforcement learning experiments. Of course, I still need to test the autoencoder in simulation before I do that.