|  |  |  |  |
| --- | --- | --- | --- |
| **MEMORANDUM** | |  | |
|  |  | | |
| **To:** | Charlie Refvem, Lecturer, Department of Mechanical Engineering, Cal Poly SLO | | |
|  | [**crefvem@calpoly.edu**](mailto:crefvem@calpoly.edu) | | |
| **From:** | Michael Shokoohi | |
| **Date:** | 10/23/2025 | | |
| **RE:** | **Mecha 02** | | |
|  |  | | |

For this assignment I chose the simplist task structure possible which included collection\_task running at a period of 50ms priority 2 and serial\_task running at a period of 100ms priority 1. The fsm for collection\_task includes states setup, idle and run while publish includes idle and publishing. The shares between tasks includes good\_to\_publish, start\_test, collecting\_data which essentially act as Boolean acknowledgements from one task to another reflecting what step they should move to. The collection has a faster period and priority because when a step response is being measured it is critical for that task to run in order to get the necessary data, while the publishing task is much lower priority because it simply empties the queues to the UART connection and doesn’t allow for another test to be conducted until it is finished anyway. The queues used are 100 elements in length holding all data for time, right and left position, right and left velocity.

A screen shot of a computer

AI-generated content may be incorrect.

Figure 1. Shown is the Task diagram used for checking user input, collecting data, and publishing it to the serial connection.

A graph with white text

AI-generated content may be incorrect.

Figure 2. Shown are the states present within the serial task

A screen shot of a black grid

AI-generated content may be incorrect.

Figure 3. Shown are the states present in the Collection Task.

|  |  |  |
| --- | --- | --- |
| Share name | Data type | Purpose |
| Good\_to\_publish | 8 bit | Allows collection task to fill all queues before letting the serial task remove and process values from the queues |
| Start\_test | 8 bit | Allows the serial task to tell collection task to begin step response test (turns true when user input/ running automated script occurs) |
| Collecting\_data | 8 bit | This share is similar to good\_to\_publish but only allows the serial task to change into the publishing state not actually publish the data. This is important because its effect allows for the serial task to stop monitoring for user input. |

A screen shot of a computer

AI-generated content may be incorrect.Figure 4. Shown is the task profile printed to console from the scheduler. Not all of my tasks have run on time but I believe that to be an artifact of the initialization/configuration setup states for each task taking longer than expected. Overall from the user perspective it does provide the expected behavior.

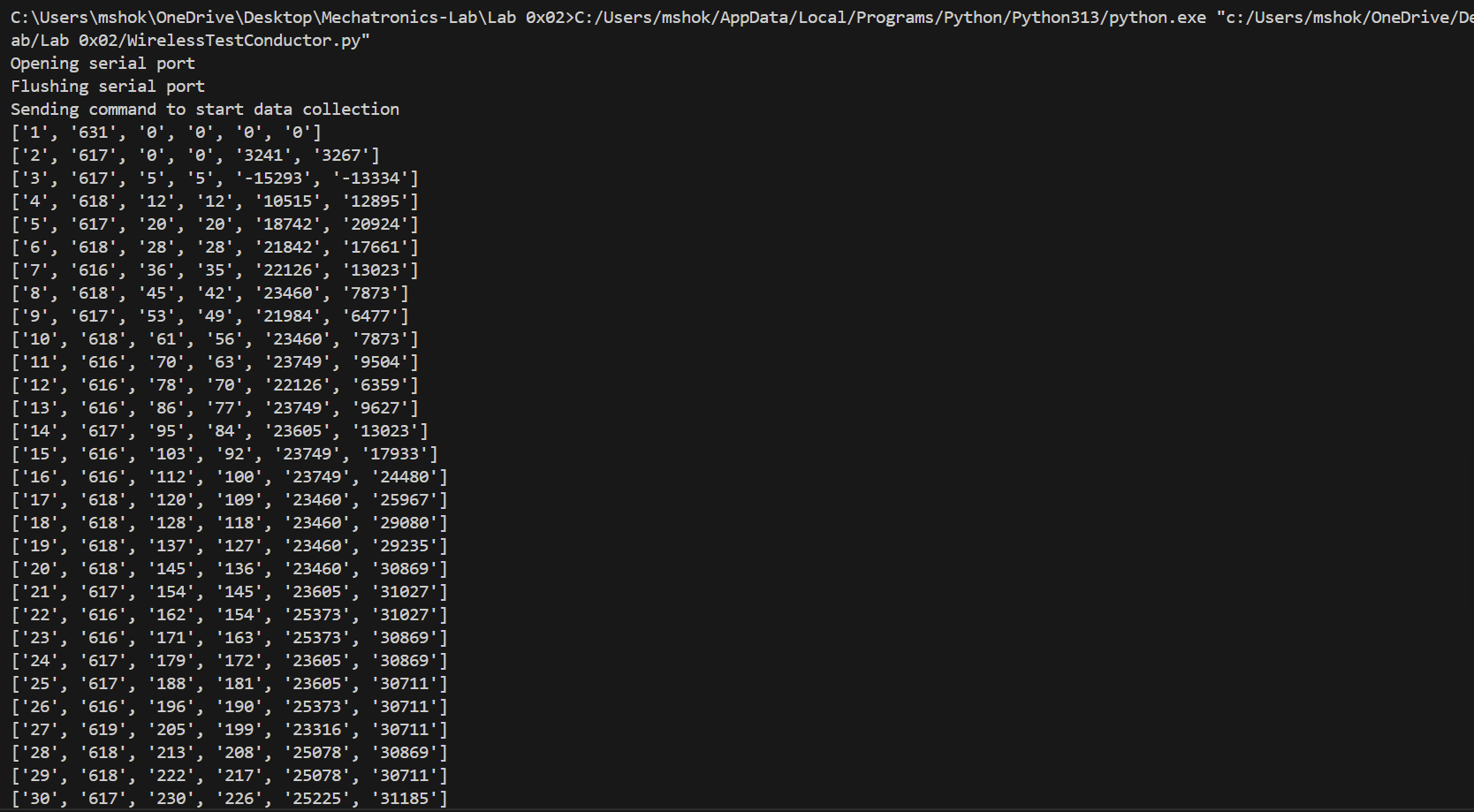


Figure 5. Shown is the data outputted from the Nucleo and received to the Bluetooth COM port on PC

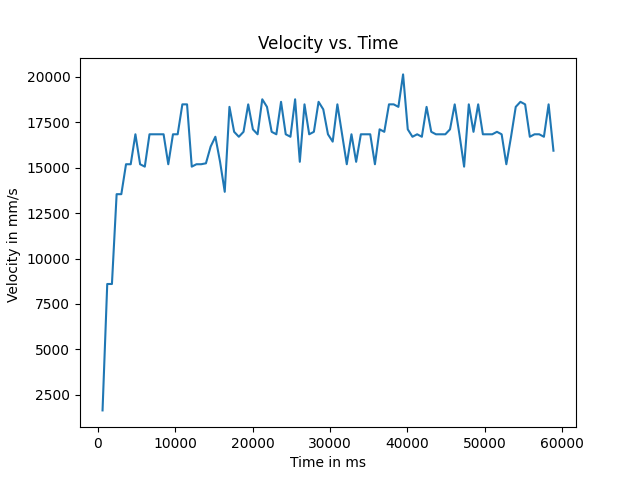


Figure 6. Shown is a plot of the data captured during a step response test

Source files used:

Motor.py, encoder.py (hardware drivers)

Main.py defines shares, queues, tasks and allows the scheduler to run

Tasks.py this is where the tasks themselves are defines along with their fsm

WirelessTestConductor.py (This is the automated PC script that runs tests and plots data)