1Preliminary Deliverable

To:	Charlie Refvem, Department of Mechanical Engineering, Cal Poly SLO crefvem@calpoly.edu	
From:	Antonio Ventimiglia ventimig@calpoly.edu	Caiden Bonney clbonney@calpoly.edu
Date:	10/2/2025	
RE:	**Lab 0x02: Preliminary Deliverable**	

Figure 1 is the task diagram created for the Lab 0x02 code.

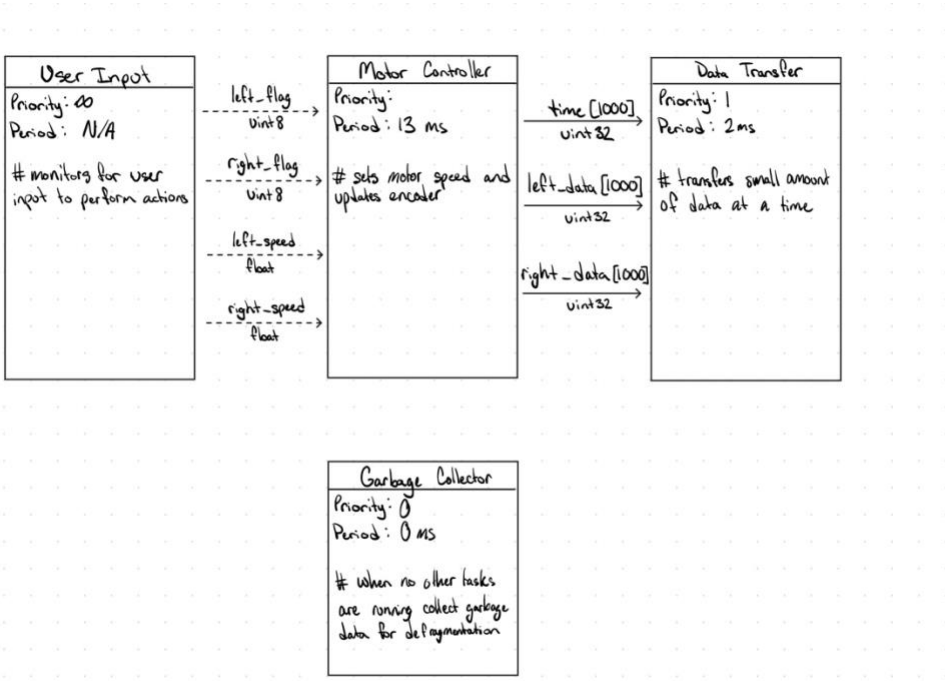

Figure 1. Task Diagram for Romi Controller

As seen in Figure 1, there are four tasks: User Input, Motor Controller, Data Transfer, and Garbage Collector. The User Input waits for the keyboard input from the user to pass along to the motor controllers. The information would be which motor is being changed, which is communicated with a Boolean named “xxx_flag” in which will be prefixed with the appropriate motor. Additionally, there will be a set speed (float) named “xxx_speed.” Both are shares. The Motor Controller task will contain the update encoder, closed loop controller, and motor actuation functionalities. The data read from both encoders will be queued to the Data Transfer, which will be named “xxx_data.” Lastly, the Garbage Collector will run and defragment data when no other actions are being done.

As seen in Figure 1, there are four tasks: User Input, Motor Controller, Data Transfer, and Garbage Collector. The User Input waits for the keyboard input from the user to pass along to the motor controllers. The information would be which motor is being changed, which is communicated with a Boolean named "_flag" in which will be prefixed with the appropriate motor. Additionally, there will be a set speed (float) named "xxx_speed."

Both are shares. The Motor Controller task will contain the update encoder, closed loop controller, and motor actuation functionalities. The data read from both encoders will be queued to the Data Transfer, which will be named "xxx_data." Lastly, the Garbage Collector will run and defragment data when no other actions are being done.