Command-and-control Subsystem for Regolith Mining Robot

CSE Member:

Pablo Canseco   pcanseco2011@my.fit.edu

Other members:

Domenick Albanese   Software
Ronald-Dean Allado   Software
Kyle Rieder   Software / Consultant

1. **Milestone 3 progress matrix**
   a. * means assistance from other team members working on the same subsystem.

Task	RasPi GPIO Manipulation	Gamepad Control	Receive RasPi Data
Completion %	80%	80%	10%
Pablo	80%	80%	10%
To do	Research more into data reception	Implement more commands. Packet framing.	Further implementation and sample data needed.

2. **Current milestone summary**
   a. ***Raspberry Pi GPIO Manipulation*** - The Raspberry Pi is now communicating with the Arduino board with a single wire over a Serial connection. The GPIO pins of the Arduino are being manipulated by the Raspberry Pi. The Python server program running on the Raspberry Pi already has the basic capability and needs to be expanded to more robot functions. The Arduino C++ firmware is ready and completely pass-through. It takes commands for what to do with its pins and does it.

b. ***Gamepad Control*** - A Sony Playstation 4 controller can now be used with the Control Station GUI to send commands to the Raspberry Pi, and in turn, the robot. The SharpDX DirectInput library was used for this. In addition, to observe competition objectives of keeping data transmission low, data reduction has been implemented such that the controller only sends data *when it changes* and restricted to discrete "levels" instead of continuously blasting numbers. Currently, the right trigger sends speed levels 1 through 5 to a motor and spins it at varying speeds with 5 being maximum speed and 1 being the slowest.

c. ***Receive Raspberry Pi Data*** - Progress on this task has been low because there hasn't been any sample data generated by the test equipment to be fetched back to the Control Station GUI. It's planned to make more progress on this for the next milestone.

3. **Milestone 4 Plan**
   a. * means assistance from other team members working on the same subsystem.

Task	Merge GUI and Controller functions	Implement Packet Framing on Raspberry Pi	Implement server-client data reception
Pablo	90%*	90%*	50%*

4. **Milestone 4 Plan Description**
   a. ***Merge GUI and Controller Functions*** - Using the controller is currently separate from using the GUI but the functionality is still in the same program. The controller should update the GUI when it's interacting with the robot. Additionally, there are intermittent crashes when either the gamepad or the network communications functionality is in a bad state. This needs fixing. Finally, it's planned to implement the rest of the controller functions instead of just a trigger.

b. ***Implement Packet Framing on the Raspberry Pi*** - Occasionally, the Raspberry Pi will receive two packets from one reading of the network socket buffer. The way incoming data is currently processed makes it so this situation leads to a crash. It's intermittent but for the sake of having robust communications that work perfectly all of the time, this needs fixing. Additionally there is a crash depending on how the client disconnects, leading to requiring a server restart in order to connect again. Disconnections shouldn't cause crashes.

c. ***Implement server-client data reception*** - This wasn't progressed upon very much on the current milestone so it's been moved to the next one. The primary reason is that there is currently no equipment received that can generate sample data to be sent back to the control station. The plan for the next milestone is to work around this by just using a random number as a "battery percentage" on the Arduino, and send it back through the Raspberry Pi and back to the control station GUI, continuously if possible.

Sponsor feedback on each task for the current Milestone

Raspberry Pi GPIO Manipulation

Gamepad Control

Receive Raspberry Pi Data

Sponsor Signature: _____ Date: _____

Sponsor Evaluation

- Sponsor: detach and return this page to Dr. Chan (HC 322)
- Score (0-10) for each member: circle a score (or circle two adjacent scores for .25 or write down a real number between 0 and 10)

Pablo	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10

Sponsor Signature: _____ Date: _____