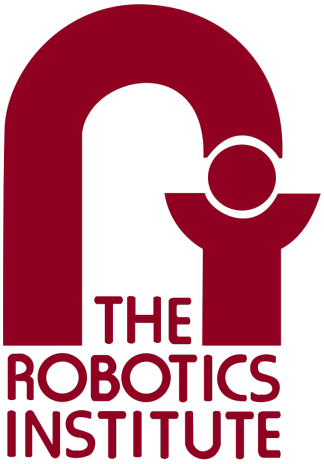
Individual Lab Report - 06



Lunar ROADSTER

Team I

Author: **Bhaswanth Ayapilla**
Andrew ID: bayapill
E-mail: *bayapill@andrew.cmu.edu*

Teammate: **Deepam Ameria**
ID: dameria
E-mail: *dameria@andrew.cmu.edu*

Teammate: **Boxiang (William) Fu**
ID: boxiangf
E-mail: *boxiangf@andrew.cmu.edu*

Teammate: **Simson D'Souza**
ID: sjdsouza
E-mail: *sjdsouza@andrew.cmu.edu*

Teammate: **Ankit Aggarwal**
ID: ankitagg
E-mail: *ankitagg@andrew.cmu.edu*

Supervisor: **Dr. William "Red" Whittaker**
Department: Field Robotics Center
E-mail: *red@cmu.edu*

September 25, 2025

Carnegie Mellon University
Robotics Institute

Contents

1 Individual Progress

1.1 Computations

1.1.1 Docker Setup on Orin

The entire docker setup on the Orin was completed over the summer by Deepam. Since I had previously configured the same environment on the Jetson AGX Xavier last semester, I supported him with technical details during the process. After returning back to Pittsburgh, I worked with Deepam to resolve outstanding issues and together we set up Docker Compose for easier orchestration. Additionally, I contributed by adding helpful aliases inside Docker and enabling ZSH features, which streamlined the workflow and made setup faster and more convenient for the team. Figure 1 shows the docker container.

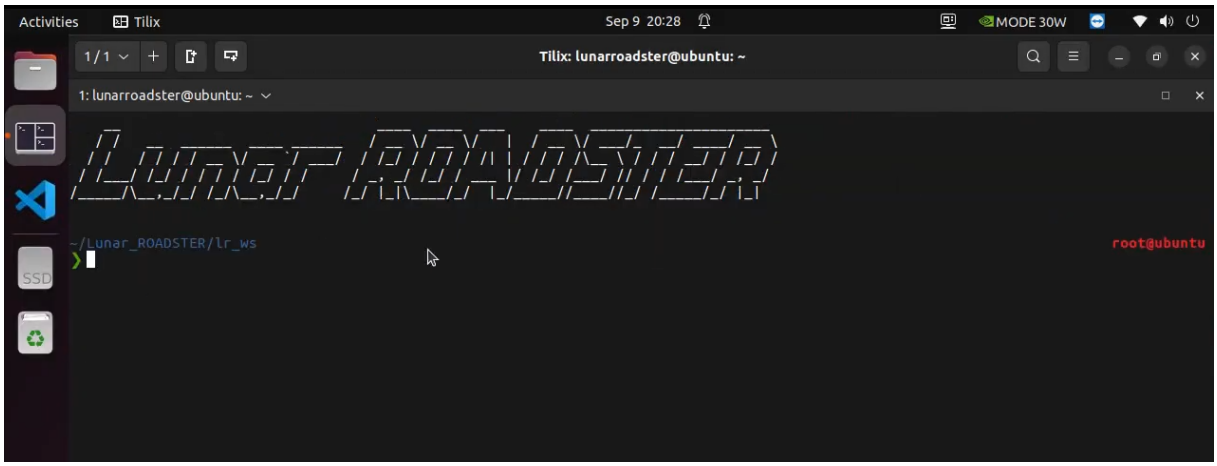


Figure 1: Docker Container

1.1.2 Code Quality Compliance

To ensure future teams can seamlessly build upon our work, we prioritized improving code documentation and standardization. As part of this effort, we added detailed descriptions for key nodes, standardized namespaces, updated header information, and revised `package.xml` metadata across packages. Over the summer, I contributed by updating several packages and building their workspaces to resolve potential issues. Each team member worked on different components and we ultimately integrated all contributions into a unified, well-documented codebase. Figure 2 shows the code description template for important nodes.

1.1.3 Master Launcher

A significant time bottleneck during our SVD preparation was the manual system bring-up. Each run required opening multiple terminals, attaching to Docker containers individually, and launching nodes in a strict order. This process was cumbersome and often delayed overall setup.

To streamline this, I developed a master launcher using Tmux, and consulted Deepam for ideas on this. I wrote a bash script that automatically opens multiple Tmux panes, attaches each to Docker, and executes the required commands. The setup provides a consolidated view of all running terminals, as shown in Figure 3. By leveraging Tmux functionality, we can easily maximize individual panes, switch seamlessly between terminals, stop or restart nodes, and terminate the entire session with a single command.

```
/**
 * @file imu_tf_publisher.cpp
 * @brief Publishes a static transform between base_link and IMU, and visualizes IMU orientation in RViz.
 *
 * This node subscribes to IMU data and continuously publishes a transform from the `base_link` to the `imu_link`
 * based on the orientation received in the IMU message. It assumes a fixed translation (zero offset), and publishes
 * a visualization marker (green arrow) in RViz for real-time orientation feedback.
 *
 * This node is useful for debugging IMU alignment and integrating orientation data into a TF tree.
 *
 * @version 1.0.0
 * @date 2025-07-07
 *
 * Maintainer: Bhaswanth Ayapilla, Boxiang (William) Fu
 * Team: Lunar ROADSTER
 * Team Members: Ankit Aggarwal, Deepam Ameria, Bhaswanth Ayapilla, Simson D'Souza, Boxiang (William) Fu
 *
 * Subscribers:
 * - /imu/data/base_link: [sensor_msgs::msg::Imu] Incoming IMU data containing orientation for the base_link frame.
 *
 * Publishers:
 * - /imu_orientation_marker: [visualization_msgs::msg::Marker] RViz marker showing orientation from IMU.
 * - TF: [geometry_msgs::msg::TransformStamped] Broadcasts a transform from base_link to imu_link using orientation from IMU.
 *
 * Services:
 * - None
 *
 * Parameters:
 * - None (uses hardcoded frame names: "base_link" and "imu_link")
 *
 * @credit Inspired by standard IMU TF and RViz marker publishers in ROS2.
 */
```

Figure 2: Code Description

This solution has significantly reduced setup time and improved system usability for the entire team.

1.2 Electronics

1.2.1 Sourcing New IMU

The rover currently uses a VectorNav VN-100 IMU for localization. However, we have consistently encountered drift issues and initial frame orientation offsets. Given that this IMU is quite old and has been used by multiple teams, we suspect potential hardware degradation.

To address this, the team decided to purchase a new IMU, as our budget allows for the upgrade. I was looking into potential IMUs to buy, shortlisted suitable options, and reached out to vendors for quotes on each. The new IMU will be ordered before PR 8 so that the team can begin setting up the new drivers too.

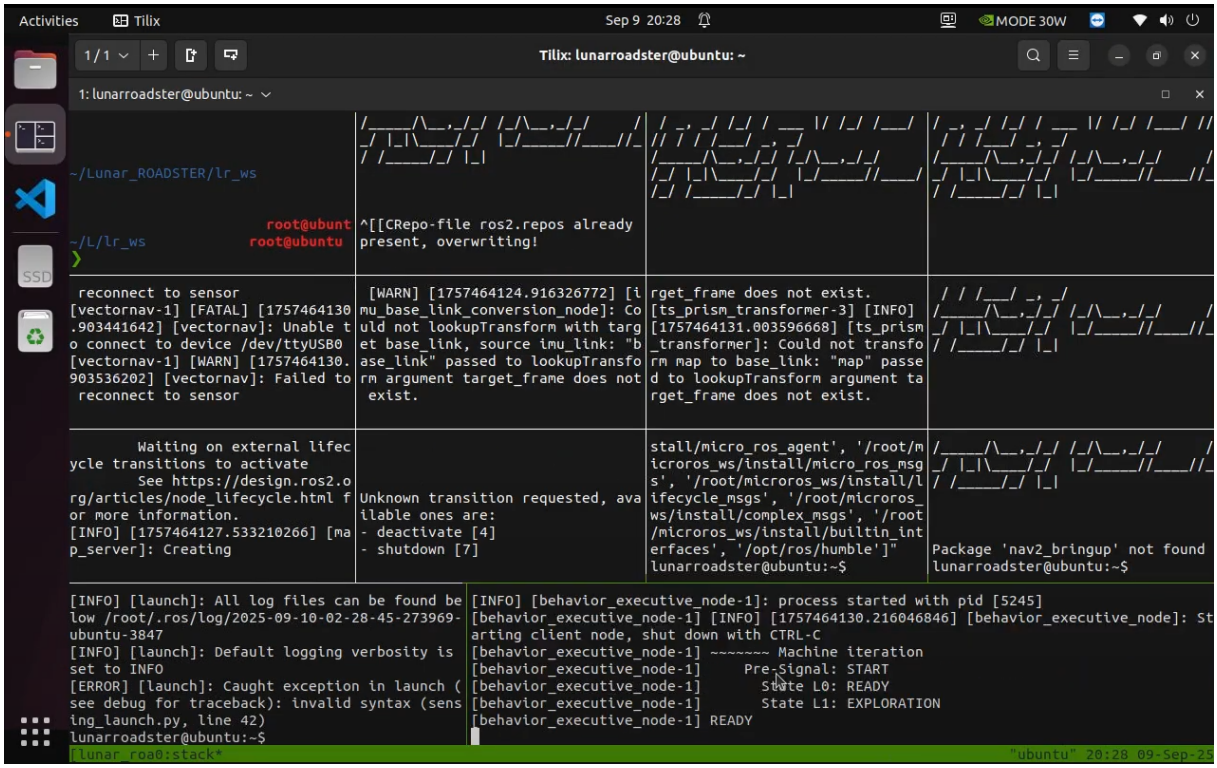


Figure 3: Tmux Master Launcher

2 Challenges

One of the key tests planned for PR 7 was a full run-through of our Spring Validation Demo. In preparation, we performed overall hardware maintenance on the rover and integrated the new onboard computer, the NVIDIA Jetson Orin. This required ensuring compatibility with not just the legacy code, but also the new namespace restructuring and updated message types, which the team worked on over the summer and as detailed in my "Code Quality Compliance" section of Individual Progress. However during testing, we encountered a critical issue where the Arduino was not communicating with the Orin.

To continue the test, we decided to switch back to the Jetson Xavier containing the old code. But the Arduino communication issues persisted, although they were less worked. This was unexpected as no wiring or code changes had been made since SVD Encore. Our current hypothesis is that the Arduino may have been damaged or that certain pin connections are incorrect.

I am actively debugging this issue in collaboration with Deepam and Ankit, and we aim to fully resolve it before PR 8.

3 Team Work

- **Bhaswanth Ayapilla:** My initial work involved collaborating with Deepam over the summer to set up Docker on the Orin. Together, we resolved several Docker-related issues, configured Docker Compose, and added useful aliases, although most of the set up was completed by him. He also supported me in developing the Master Launcher. Currently, I am working closely with Ankit and Deepam to debug the Arduino communication issues. Beyond this, the entire team collaborated on systems engineering discussions, brainstorming ideas for planning and improvements, and running maintenance tests on the rover in the Moon Yard.

- **Ankit Aggarwal:** Ankit's work was carried out in collaboration with Simson and Deepam to resolved hardware issues. This involved complete maintenance of the wiring connections and re-routing them in a safe and efficient configuration. Additionally, he worked on the designing mounts for the Orin and the Zed 2i camera. He also working with Deepam and I to debug the Arduino communication issues. Beyond this, the entire team collaborated on systems engineering discussions, brainstorming ideas for planning and improvements, and running maintenance tests on the rover in the Moon Yard.

- **Deepam Ameria:** Deepam's inital work was in collaboration with me over the summer, we worked on setting up the new Docker on the Orin. We worked on solving several dependency and version issues. He also worked on integrating the ZED Camera (SDK and ROS2 Wrapper) with the new Orin. He also worked with Simson and Ankit to carry out the hardware maintenance of the rover. They worked on replacing damaged wires, rerouting them, and soldering the connections to make them more robut. Additionally, he is working with me and Ankit to solve Arduino communication issues. Moreover, he collaborated with the team on systems engineering discussions, brainstorming ideas and plans for upcoming semester and conducting maintenance tests on the rover in the Moon Yard.

- **Simson D'Souza:** Simson's work was carried out in collaboration with Deepam and Ankit to identify and resolve hardware issues. This primarily involved replacing damaged wires, rerouting them for a cleaner setup, and soldering connections to ensure reliability during transportation. In addition, he worked on the setup of the graphical user interface (GUI); an initial version of the dashboard has been created and will be updated as new modules are integrated. He also contributed to the collaborative effort of refactoring the codebase and conducting rover maintenance tests.

- **Boxiang (William) Fu:** William's work since the last progress review focused on code quality & architecture improvements, and project management work. Code quality improvements include improving code readability and quality, and architecture improvements include refactoring code to conform with a Directed Acyclic Graph. Project management work include implementing the Objectives & Key Results (OKR) framework and setting up GitHub code version control.

All my goals involve not just code development, but also deploying them on the robot and testing them. So every step involves continuous coordination with different teams members and working together to test in the Moon Yard.

4 Plans

The following are my goals for progress review 8:

1. Fix arduino connection and reset issues
2. Implement a new total station resection method to fix localization offset issues
3. Begin implementing the global path planner
4. Begin implementing the global navigation controller along with Simson