Helelani Rover Overview

1. Drive Platform

The 'Alpha' Argo Rover is a ruggedized rover platform with differential drive steering. Wheels are fitted with tubeless pneumatic tires, and linked in pairs via chain-drive transmission. The entire drive platform runs on (2) 48V DC motors, located at mid-starboard and mid-port.

2. Sensors and Instruments

Helelani features a host of peripherals added by PISCES. These sensors and instruments provide useful feedback of the rover's environment and motive state.

- Color IP Cameras
- Inertial Measurement Unit (IMU)
- GPS Receiver

3. Communications

Helelani uses an IP network to transfer commands, telemetry, and camera video. There are 2 ethemet switches on board the rover to support this network.

One switch is located in the PISCES avionics box near the computer. This switch
handles communication between the computer, *Bullet* wireless antenna, and another switch located on the mast. Guest payload located in or near the avionics box may attach devices to this switch to share the JP network.

A second switch is located on the mast. This is a PoE switch and injects 48V DC into
all devices attached to it. Currently, it's only used by the 3 JP cameras mounted on the mast. Guest devices located on the mast, or devices requiring 48V PoE may connect to this switch.

The rover's uplink is in the form of a *Bullet* antenna. This omnidirectional antenna establishes a wireless ethemet link which remains stable up to ~1 km line-of-sight. A matched *Bullet* antenna is located at the command base, and wirelessly joins client computers onto the same IP network as the rover's onboard systems.

4. Onboard Computer

All onboard processing occurs within an Intel NUC small-form PC. The PC consists of a single board mounted to the side in the avionics box, leaving the bottom of the box open for additional payload. A 120 GB SSD is directly attached to the PC for storing mission data.

The PC runs Ubuntu 14.04 with rover software built on top of the Robotics Operating System (ROS) Indigo architecture. The software is divided into several 'nodes' to reflect the modular nature of the Rover's systems. So far, Helelani uses separate ROS nodes for *navigation, telemetry, camera control, SIS capture,* and *stereo image processing.*

4.1. Navigation

The primary function of the Navigation node is to control the PWM hardware to set throttle values. The node can either receive real-time differential throttle values, or discrete commands that are fulfilled using a queue. The discrete commands accepted by the node have the following grammatical structure:

- Straight line movement **drive**
- Direction **forward|backward**
- Value
- Unit **seconds|rotations|meters**
- Optional throttle value (0-100)
- Tum in place **turn**
- Direction **leftlright**
- Value
- Unit **seconds|rotations|degrees**
- Optional throttle value (0-100)
- SADL actuation **sadl**
- Direction **up|down|autolevel**
- Value (seconds for **upldown,** degrees for **autolevel)**
- Halt all motion **kill**

4.2. Telemetry

The telemetry node contains code for interfacing with each specific instrument on board Helelani. The node builds an aggregate of data from the *IMU, motor controller,*

ADC, and *GPIO* hardware components.

5. Client Software

The Helelani client software also runs on top of the ROS framework. Presently, this requires running within Ubuntu, which may be accomplished using a virtual machine such as VirtualBox. The client GUI is divided into several ROS nodes, like the onboard software. With the ROS' rqt application, each of these nodes may be docked together into a complete user interface. Different mission requirements may call for alternate GUI perspectives. This is especially useful for multiple users on different computers performing different mission roles.

Nodes can be opened through the dropdown tool menu on the GUI. Available nodes are:

- Mission Control
- IMU View
- Command
- Drive Trackpad
- Joystick Controls
- IP Camera Viewer
- Camera Control