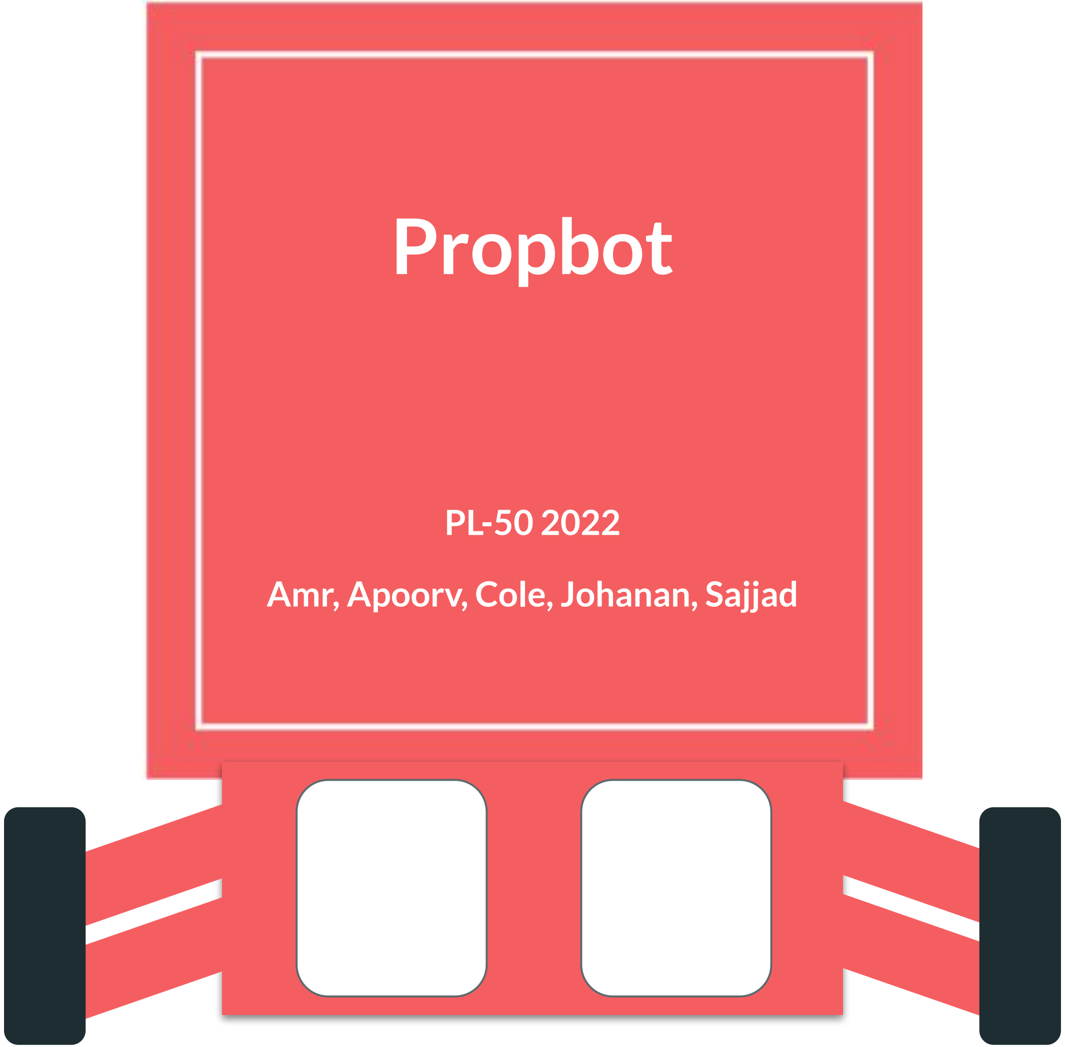


Propbot

PL-50 2022

Amr, Apoorv, Cole, Johanan, Sajjad

Overview

- Client Introduction
- Problems with data collection
- Goal
- Requirements/Constraints
- Stakeholders
- Solution
- Our contribution
- Engineering Design

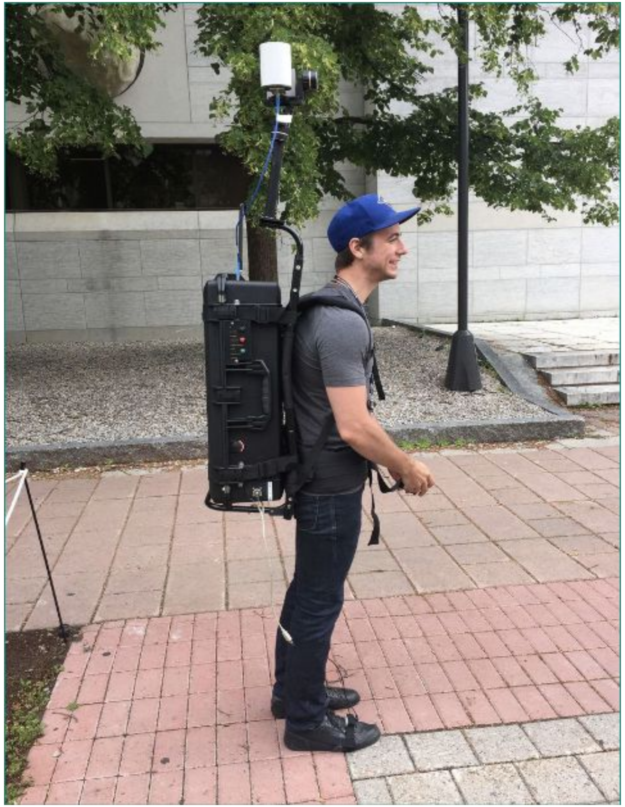
Client

- Conducts research on next-generation wireless communications and radar systems
- Requires high-quality wireless propagation data at various locations on UBC campus
- Researchers have a background in electrical and computer design

UBC Radio Science Lab

Problem with Data Collection

- No standardized commercially available solution
- RSL Researchers need to manually transport heavy and expensive equipment
- Tedious and time consuming
- Affects many researchers around the world

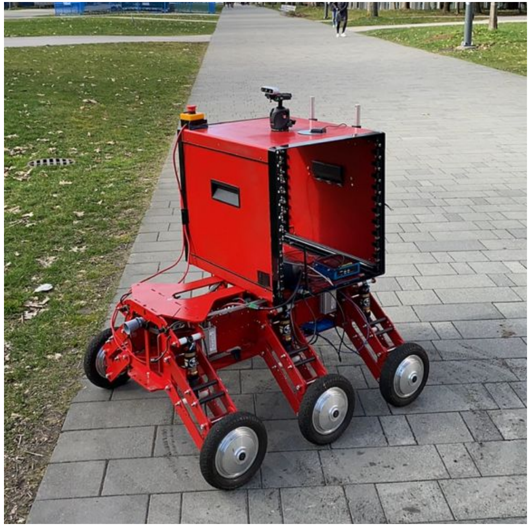


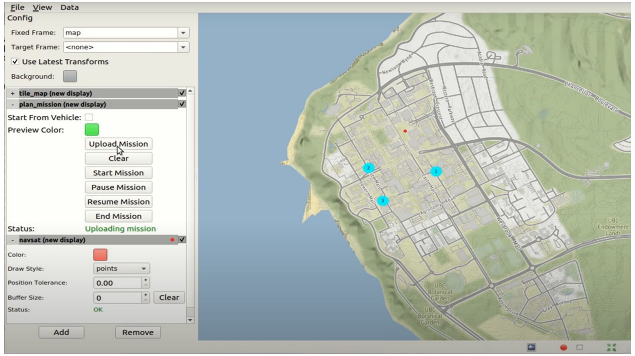
Goal

- Automate the process of collecting wireless propagation data

- Autonomously navigate to points of interest set by a user

- Traverse UBC campus safely without posing any risk to itself or the public





Config

Fixed Frame: map

Target Frame: <none>

☑ Use Latest Transforms

Background:

+ tile_map (new display) ☑

- plan_mission (new display) ☑

Start From Vehicle: □

Preview Color: 🟩

Status: Uploading mission

- navsat (new display) ● ☑

Color: 🟥

Draw Style: points

Position Tolerance: 0.00

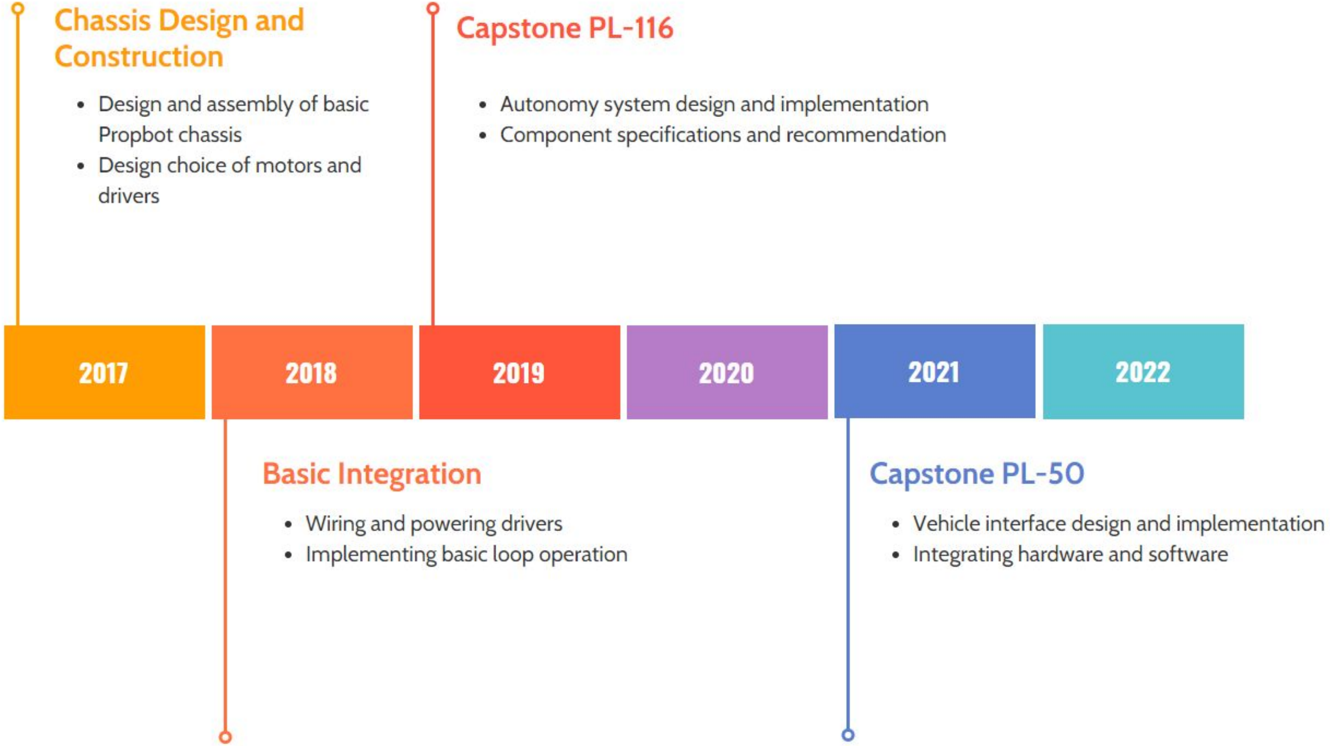
Buffer Size: 0

Status: OK

Stakeholders

UBC RSL	User	Collect propagation data and GPS position
UBC Road Users	Community	Maintain safe distance and avoid collisions
Robot Operator	User	Teleoperation control and autonomy input
Transport Canada	Legal Personnel	Abide by legal guidelines
Future Capstone Teams	Developer	Clear and understandable documentation for future teams

Propbot’s Initial State



2017 | 2018 | 2019 | 2020 | 2021 | 2022

Chassis Design and Construction

- Design and assembly of basic Propbot chassis
- Design choice of motors and drivers

Basic Integration

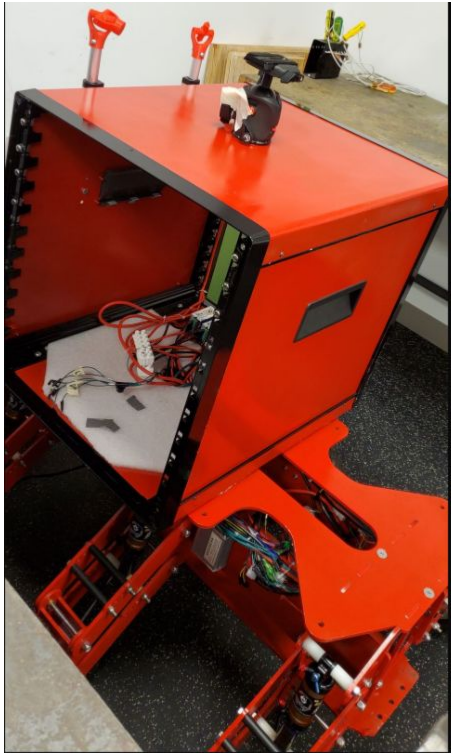
- Wiring and powering drivers
- Implementing basic loop operation

Capstone PL-116

- Autonomy system design and implementation
- Component specifications and recommendation

Capstone PL-50

- Vehicle interface design and implementation
- Integrating hardware and software



Constraints

- Mechanical Structure: chassis, suspension, rack module, hub motors

- Design recommendations: Ultrasonic sonar sensors, Depth camera, GPS module and LiDAR

- Autonomy Software: use of ROS framework for software (impacts how devices need to be interfaced), Move base for path planning.

Requirements

Motion: Execute longitudinal and rotational movements

Navigation: Facilitate motion commands from both a remote operator through RC and locally generated commands from the autonomous computer

Autonomy: Take a set of way points and autonomously generate a path of motion through the points that avoids obstacles

Safety: Must be safe and avoid/prevent collisions with objects and pedestrians

Design Principles

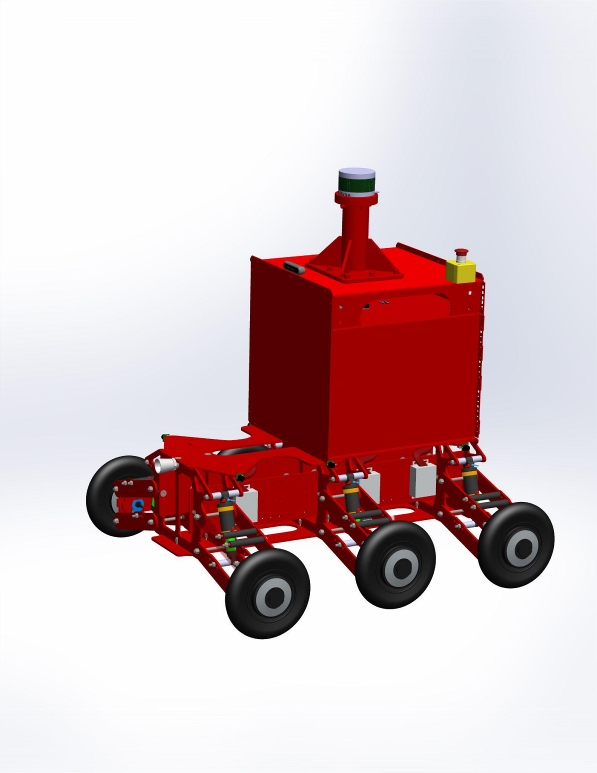
- **Safety:** Ensures safe operation for users and bystanders
- **Inheritable:** Future teams will not have to go through the same struggles we faced
- **Extendibility:** Features can be added without changing existing implementation
- **Testability:** Components and Features can be verified and validated according to known expectations

Our Contribution

Safe and reliable teleoperation

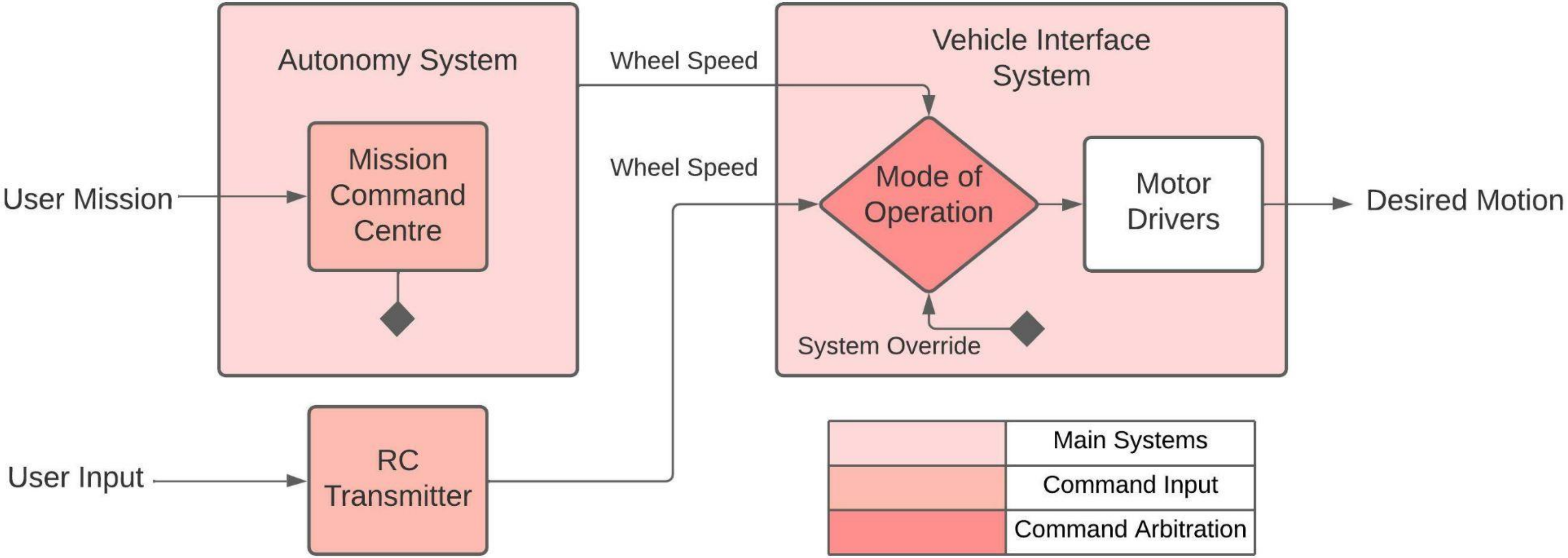
Autonomy system integration with sensor suite

Robust power distribution system

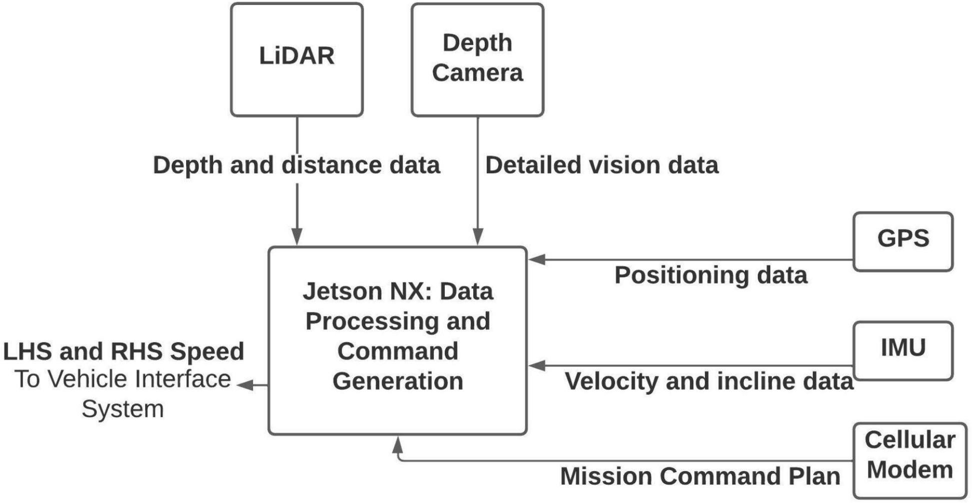


https://www.youtube.com/watch?v=06ADRLlokvQ

High-level System Overview



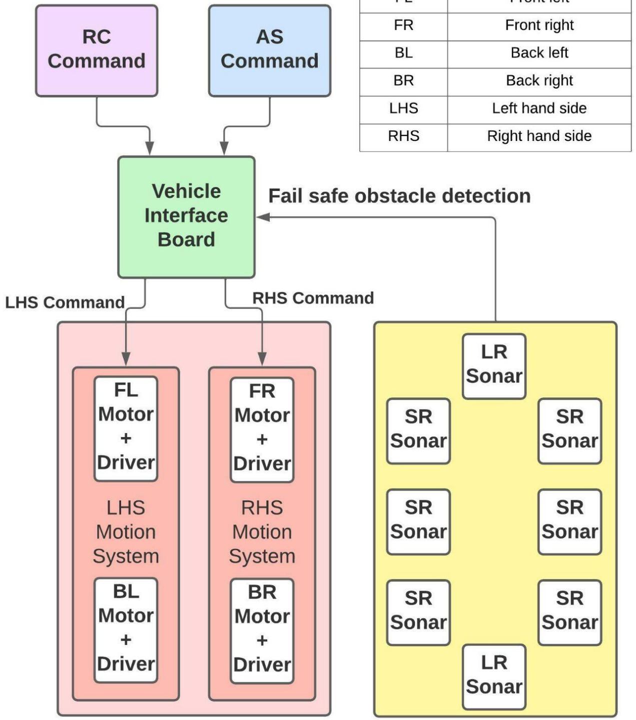
High-level Overview



Autonomy System Overview

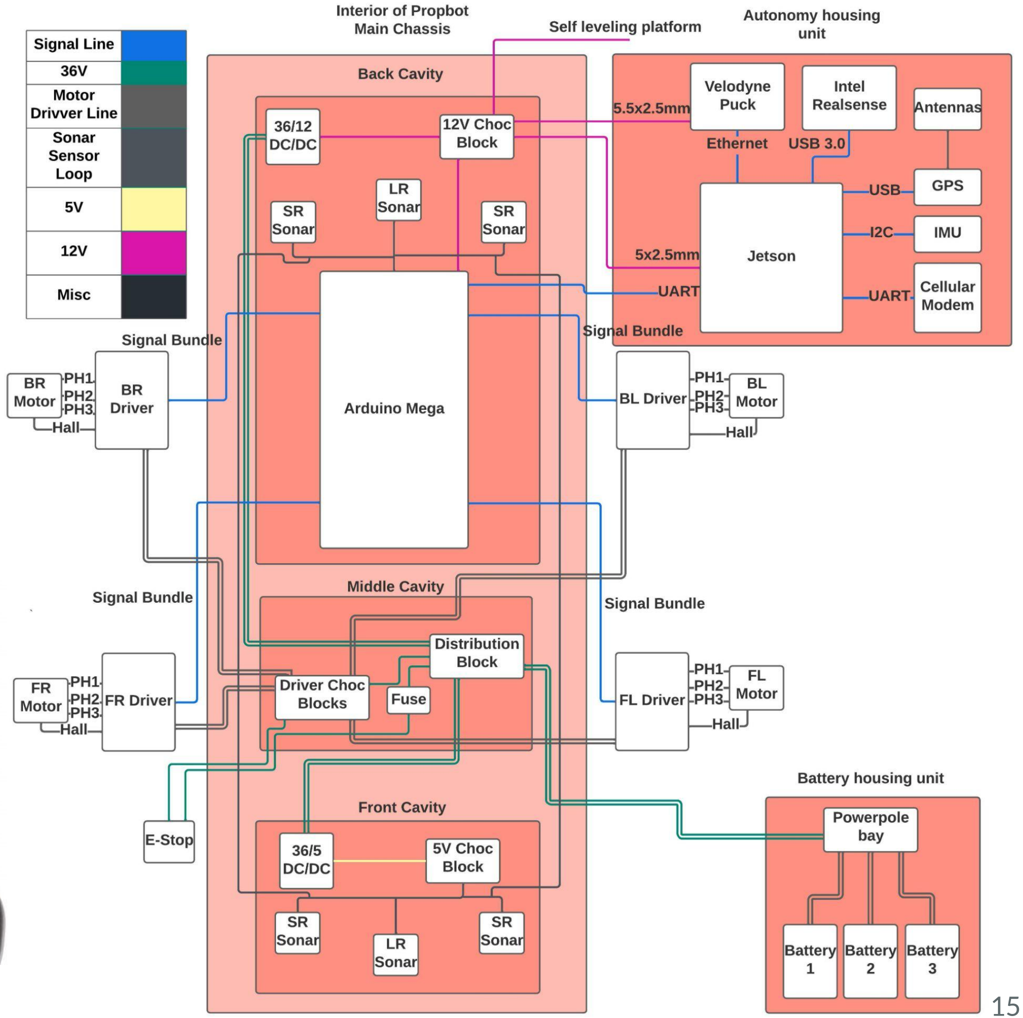
Vehicle Interface System Overview

Legend	Definition
🟪	Remote Control
🟦	Autonomy System
🟩	Command Decision
🟨	Fail Safe Object Detection
🟥	Motor Control
LR	Long-range
SR	Short-range
FL	Front left
FR	Front right
BL	Back left
BR	Back right
LHS	Left hand side
RHS	Right hand side

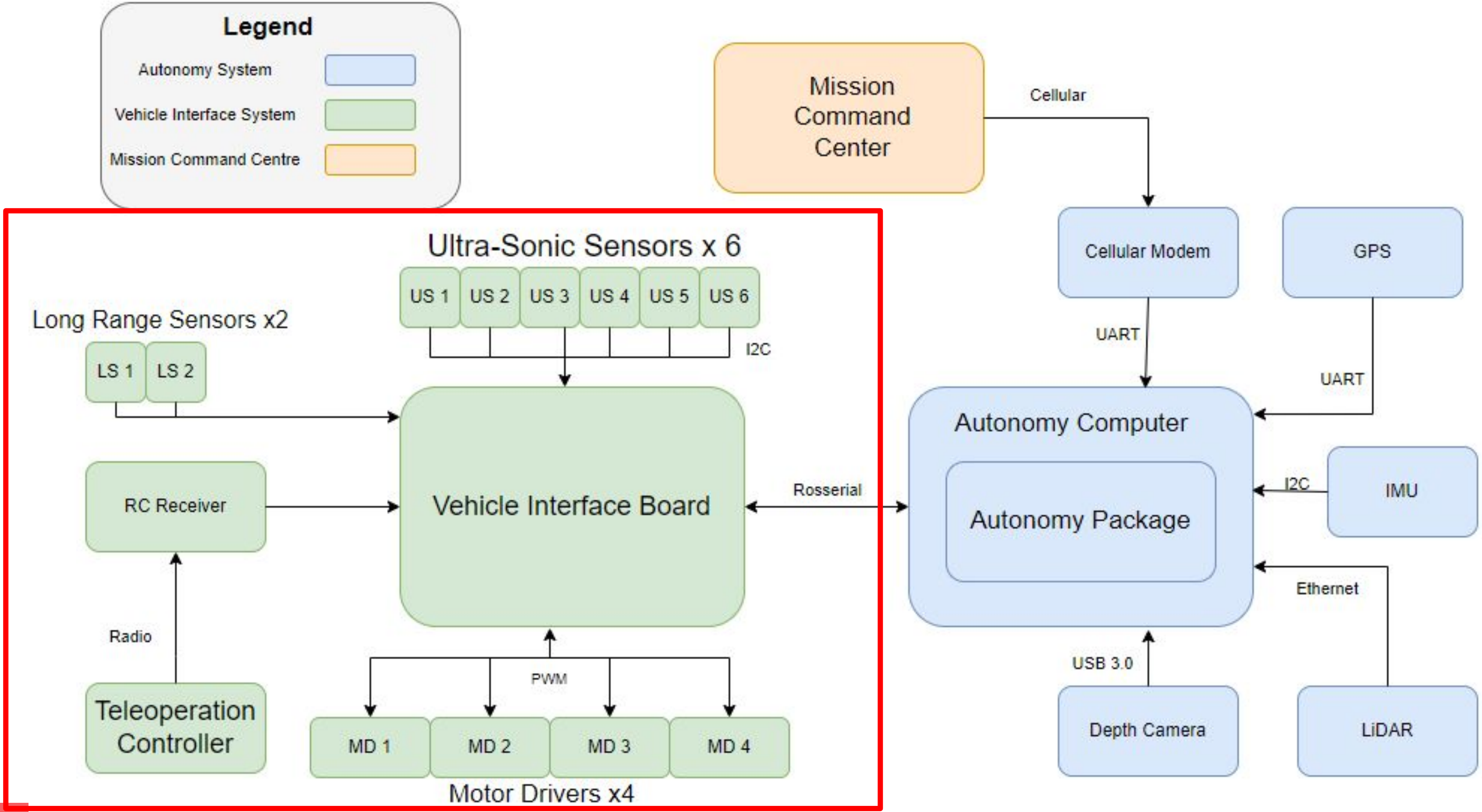


Power Management

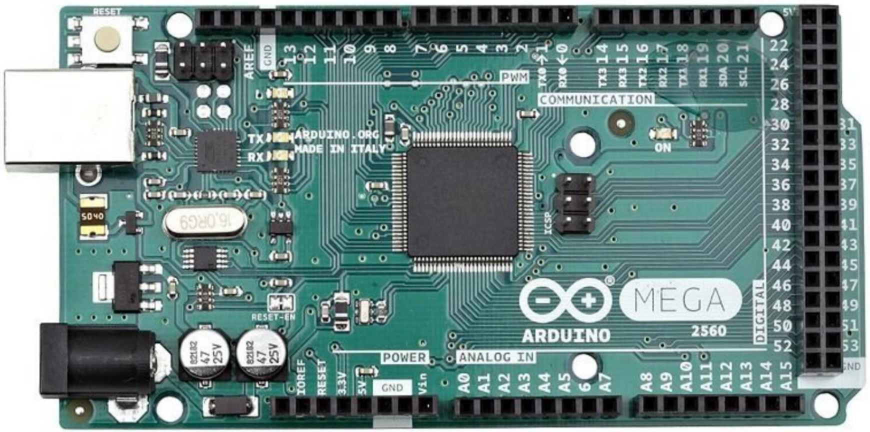
- Requirement is to provide the correct power to all components for at least 1.5hrs



Vehicle Interface System



Vehicle Interface Board



Functionality

- **Motor Control**: Propbot should be able to control the wheels
- **Teleoperation**: Propbot should be operable via RC transmitter
- **Obstacle Avoidance:** Propbot should have the capability to avoid obstacles

Motor Drivers

<u>Roboteq SBL1360A (Proposed)</u>

- Supports closed loop feedback, E-braking, CANBus
- Highly configurable via software and custom scripting

<u>Generic motor controller (Legacy)</u>

- Implemented as a risk mitigation step
- Supports basic speed, direction, and braking functionality
- Used for E-bikes with physical throttle but repurposed for Arduino PWM output using RC Filter

Sonar Sensors - Maxbotix MB1202/MB7040

MaxSonar MB1202
Wide beam, short distance

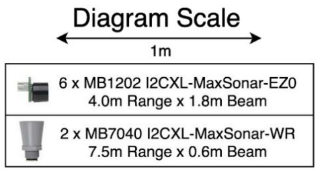


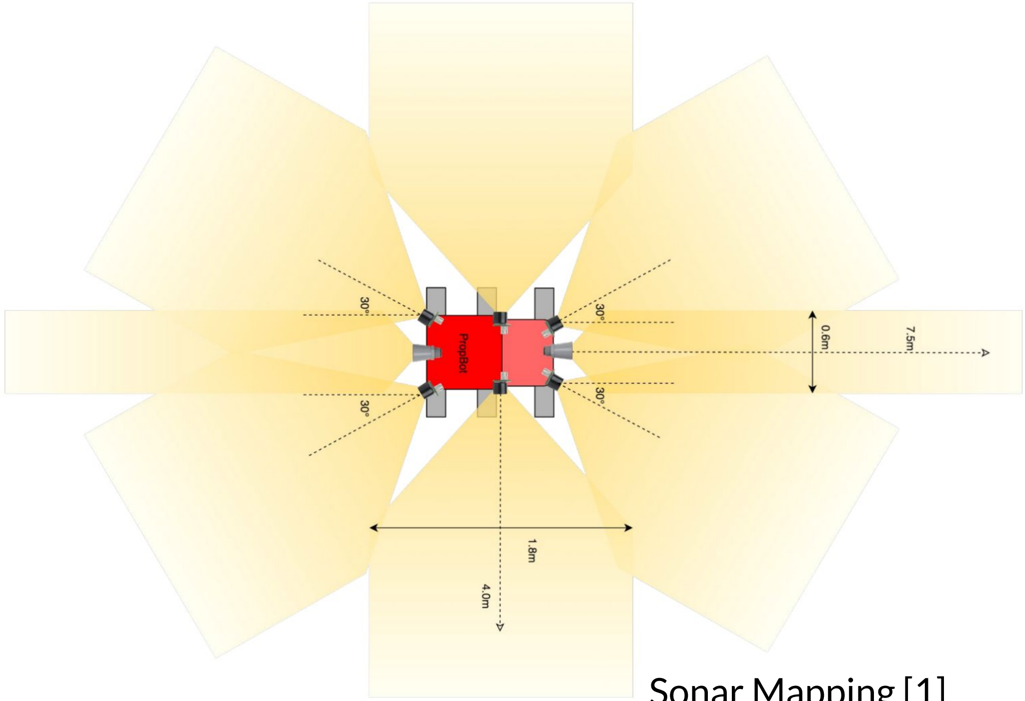
Diagram Scale

1m

6 x MB1202 I2CXL-MaxSonar-EZ0
4.0m Range x 1.8m Beam

2 x MB7040 I2CXL-MaxSonar-WR
7.5m Range x 0.6m Beam

MaxSonar MB7040
Narrow beam, long distance

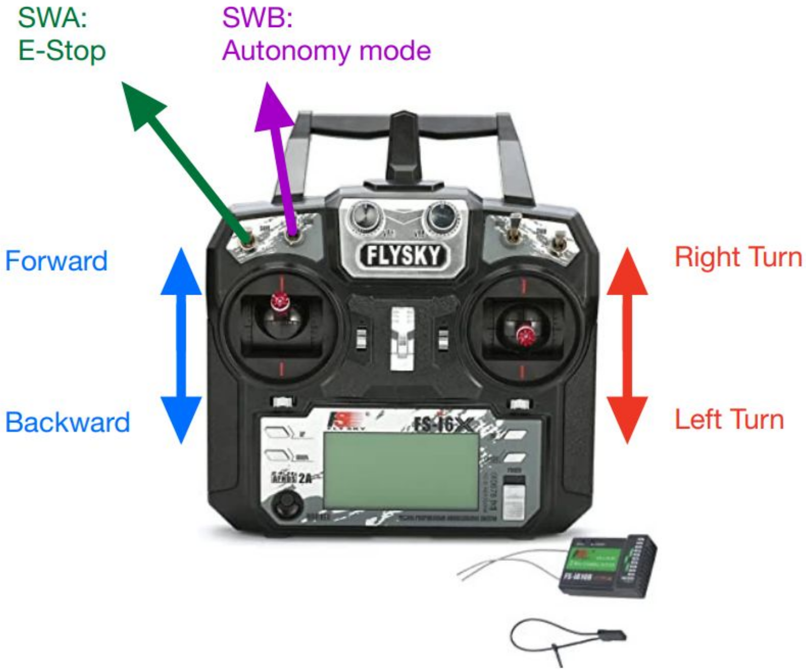


Sonar Mapping [1]

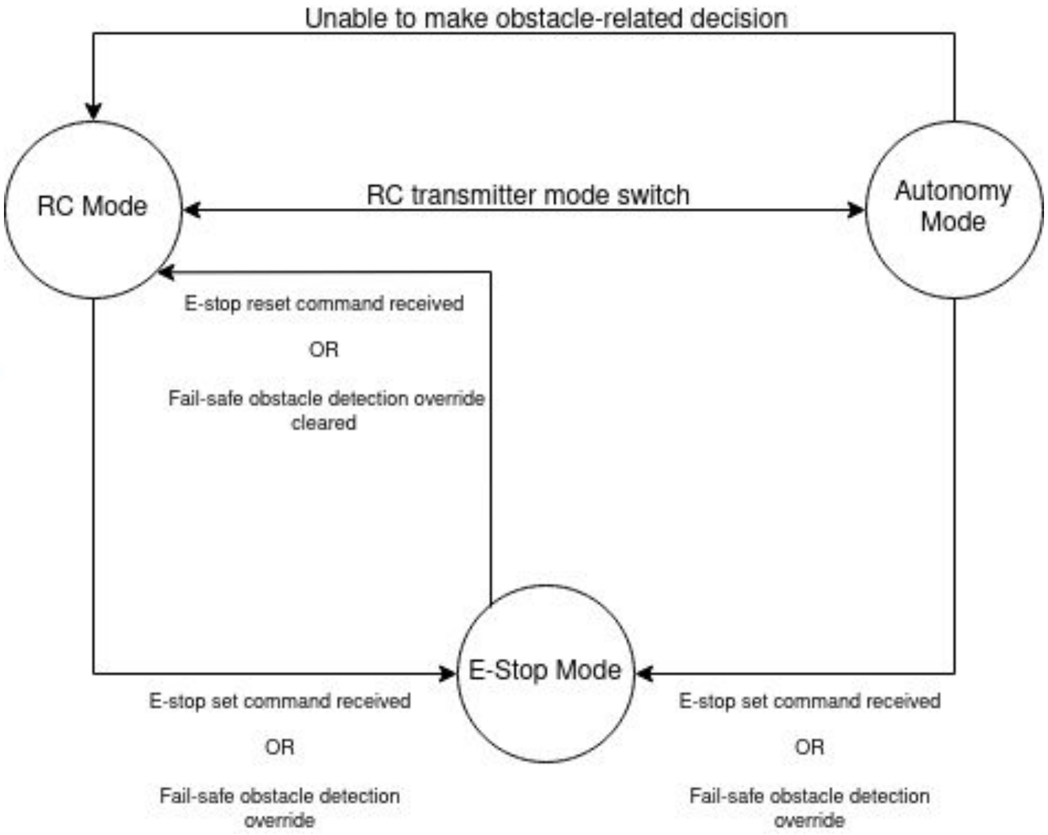
Sonar (Safety) Demo

https://www.youtube.com/watch?v=Bj1LURdYbMI

Teleoperation Control



FlySky i6X Radio Transmitter



RC Mode

Autonomy Mode

E-Stop Mode

Unable to make obstacle-related decision

RC transmitter mode switch

E-stop reset command received

OR

Fail-safe obstacle detection override cleared

E-stop set command received

OR

Fail-safe obstacle detection override

E-stop set command received

OR

Fail-safe obstacle detection override

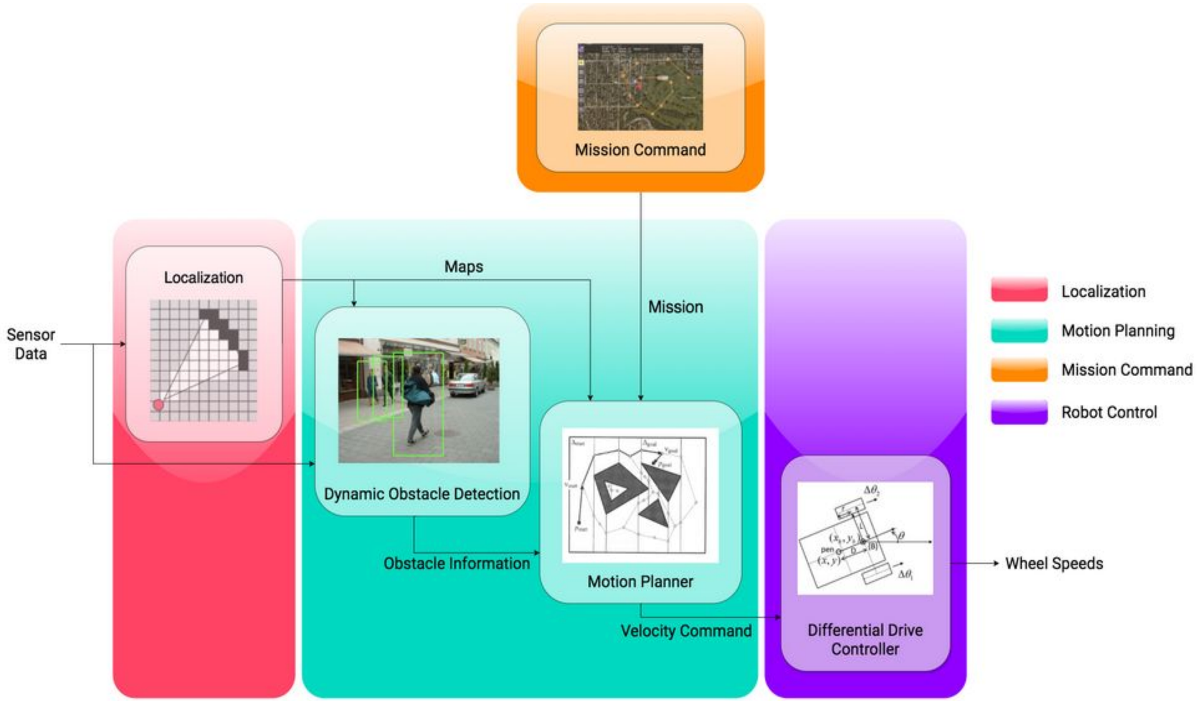
Teleoperation Demo

https://www.youtube.com/watch?v=AlADD6odR2I

Autonomy System

Requirements

- **Sensor Integration**: Interfaces with the sensors required to achieve localization and dynamic obstacle detection
- **Control**: Propbot should execute commands as issued by the Autonomy System



Autonomy Overview [1]

Autonomy System - Sensor Integration

Nvidia Jetson NX

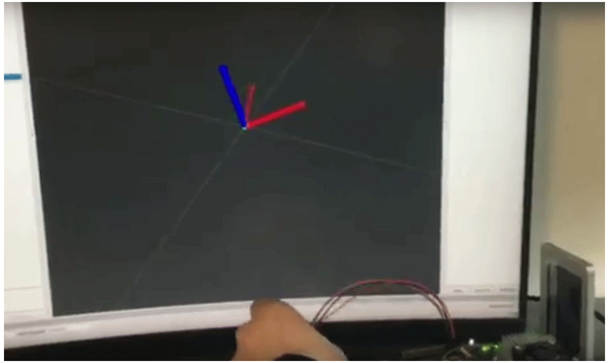
Sensor	Purpose	Communication Protocol
LiDAR: Velodyne Puck	Localization and Mapping, Navigation	Ethernet
GPS: Neo-M8P	Global Positioning, Navigation	Serial
IMU: BNo055	Localization	i2c
Depth Camera: Intel RealSense D435	Object Detection, Visual odometry (speed estimation)	Serial

Not Integrated

Integrated

Autonomy System - Sensor Integration

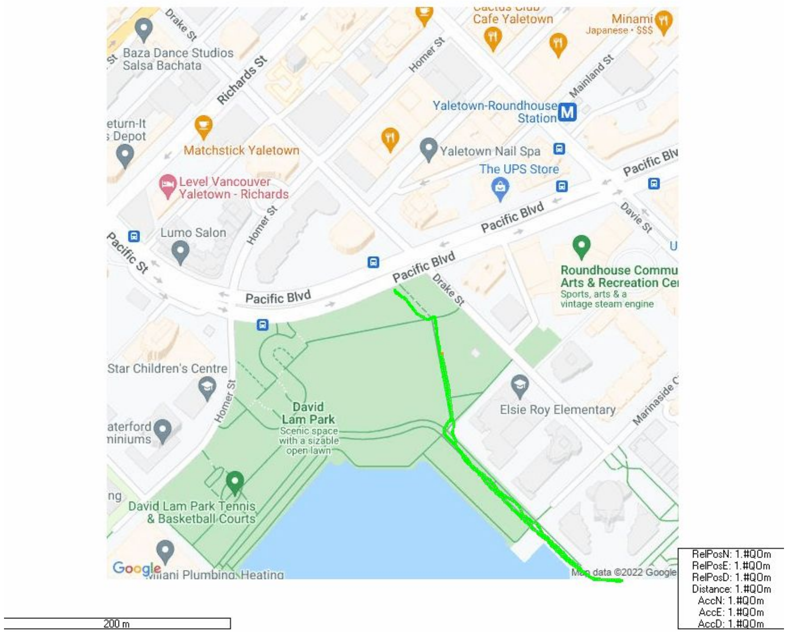
IMU



https://www.youtube.com/watch?v=TuYOaY9iNyU

Camera

https://www.youtube.com/watch?v=m4SNWCZo97o

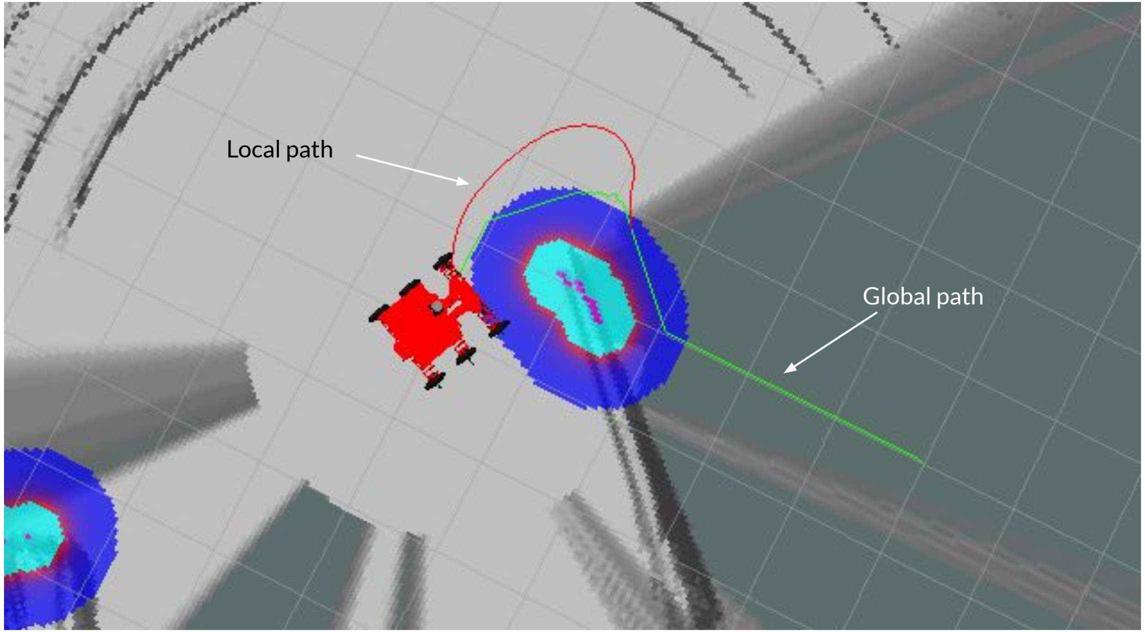


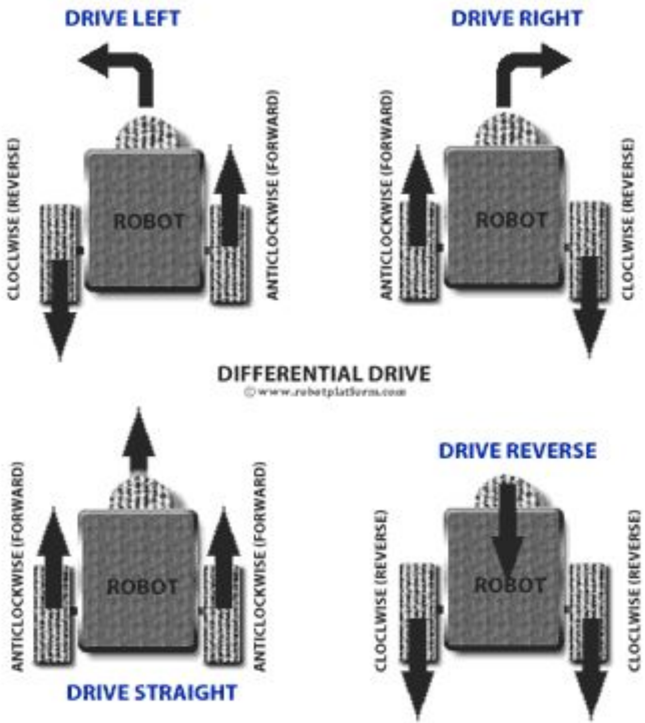
```
RelPosN: 1.#Q0m
RelPosE: 1.#Q0m
RelPosD: 1.#Q0m
Distance: 1.#Q0m
AccN: 1.#Q0m
AccE: 1.#Q0m
AccD: 1.#Q0m
```

GPS

https://www.youtube.com/watch?v=uRIkpnUqnPM

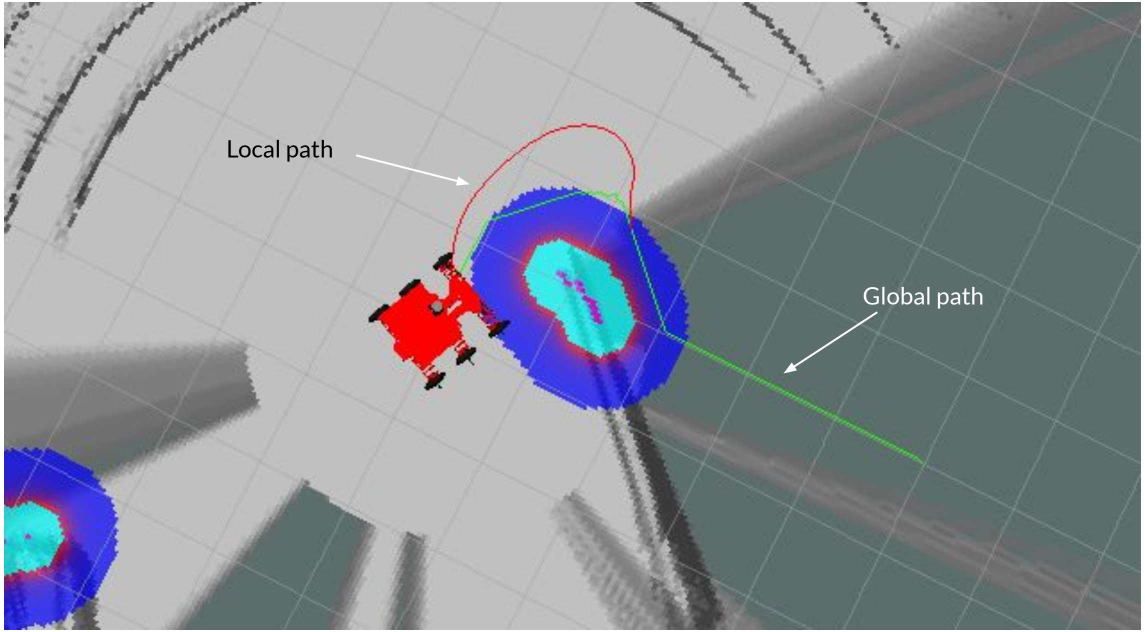
Autonomy System - Control

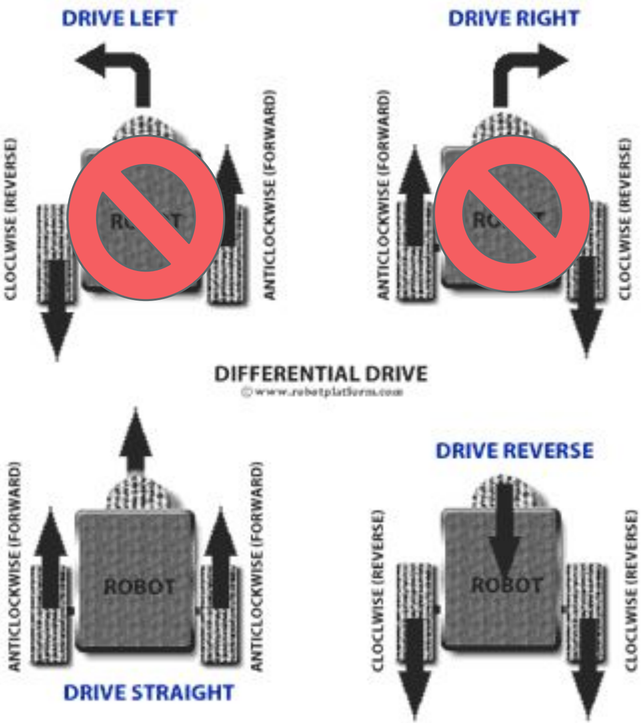




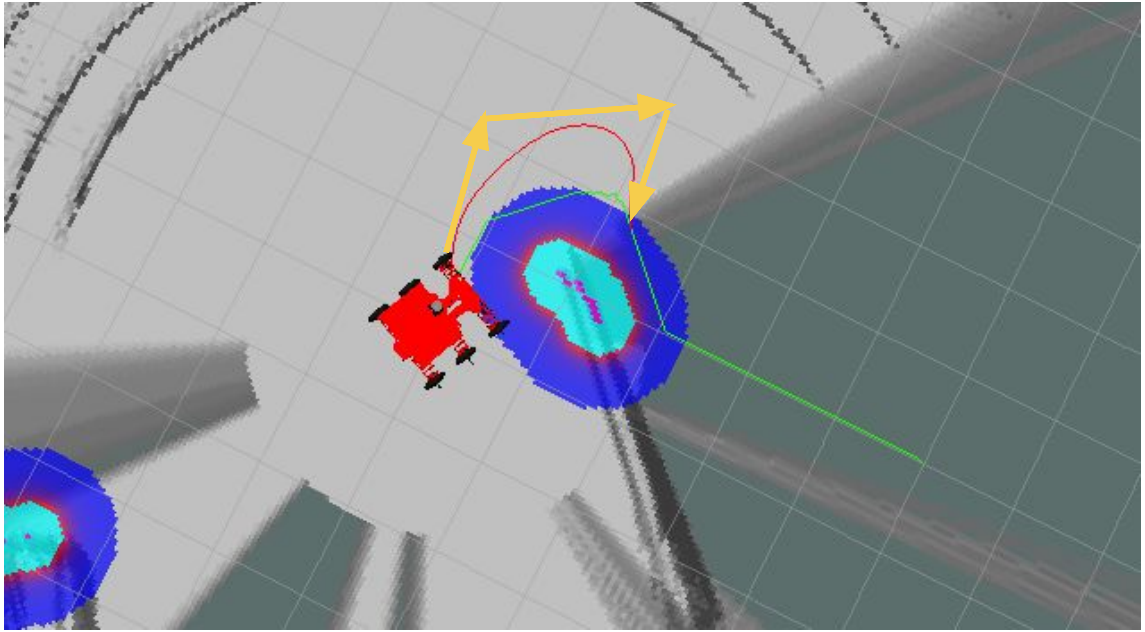
Differential Drive [1]

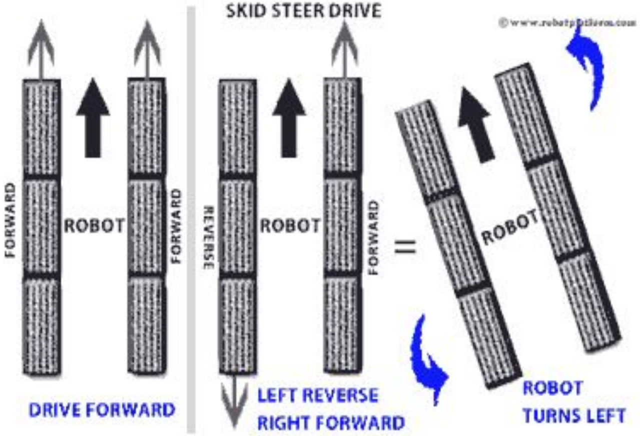
Autonomy System - Control





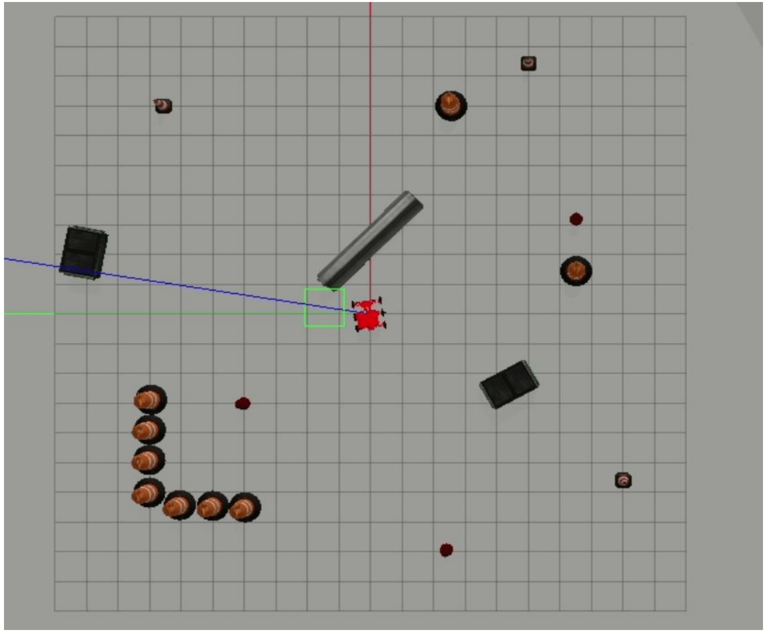
Autonomy System - Control





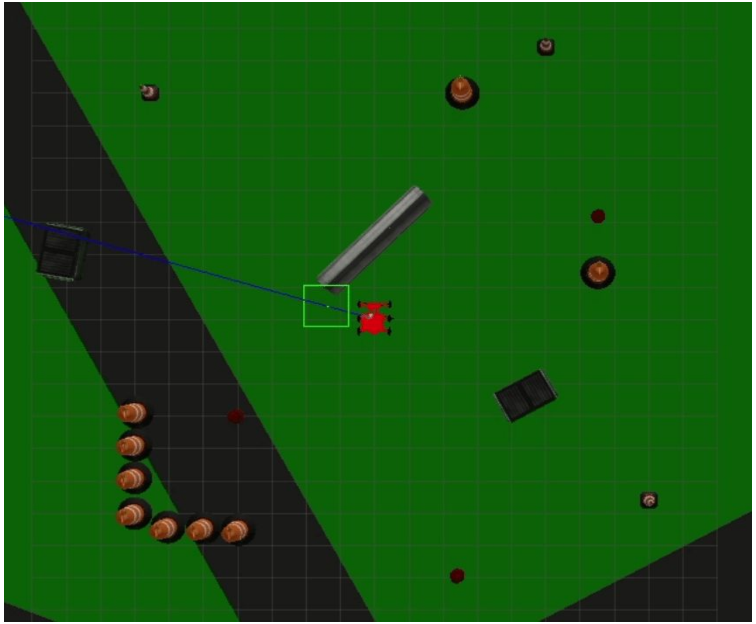
Skid Steer Drive [2]

Autonomy System - Comparing Drive Systems



Differential Drive

https://www.youtube.com/watch?v=gSyPa_sVO0w



Skid Steer Drive

https://www.youtube.com/watch?v=KRZztzgfEqc

Autonomy System - Integration

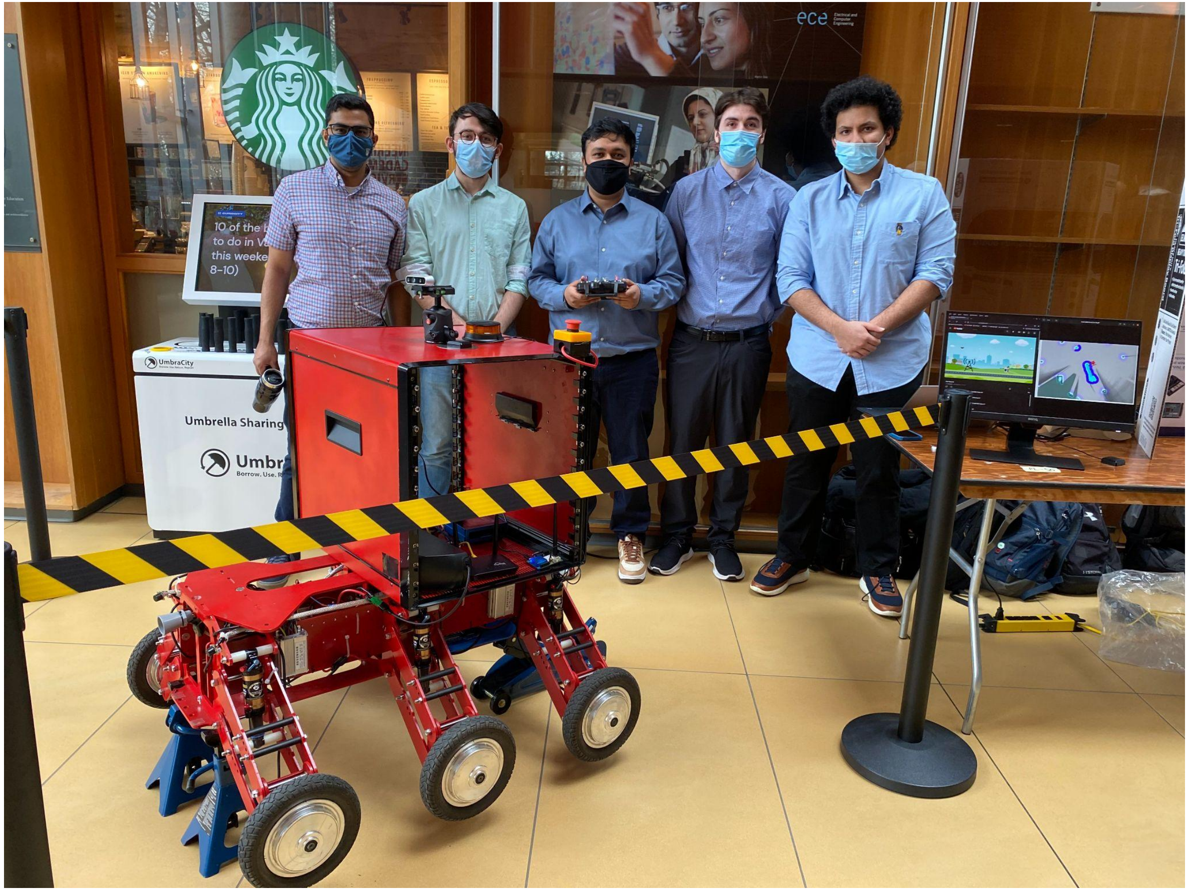
https://www.youtube.com/watch?v=fy8VR3S_pwQ

https://www.youtube.com/watch?v=qEOlnOn-JPA

Future Work

- Complete integration of LiDAR sensor
- Add wheel encoder or hall sensor for speed feedback
- Fine tuning Autonomy System configurations based on sensor input
- 3D mounts for the existing sonar sensors
- Weatherproofing

Thank you



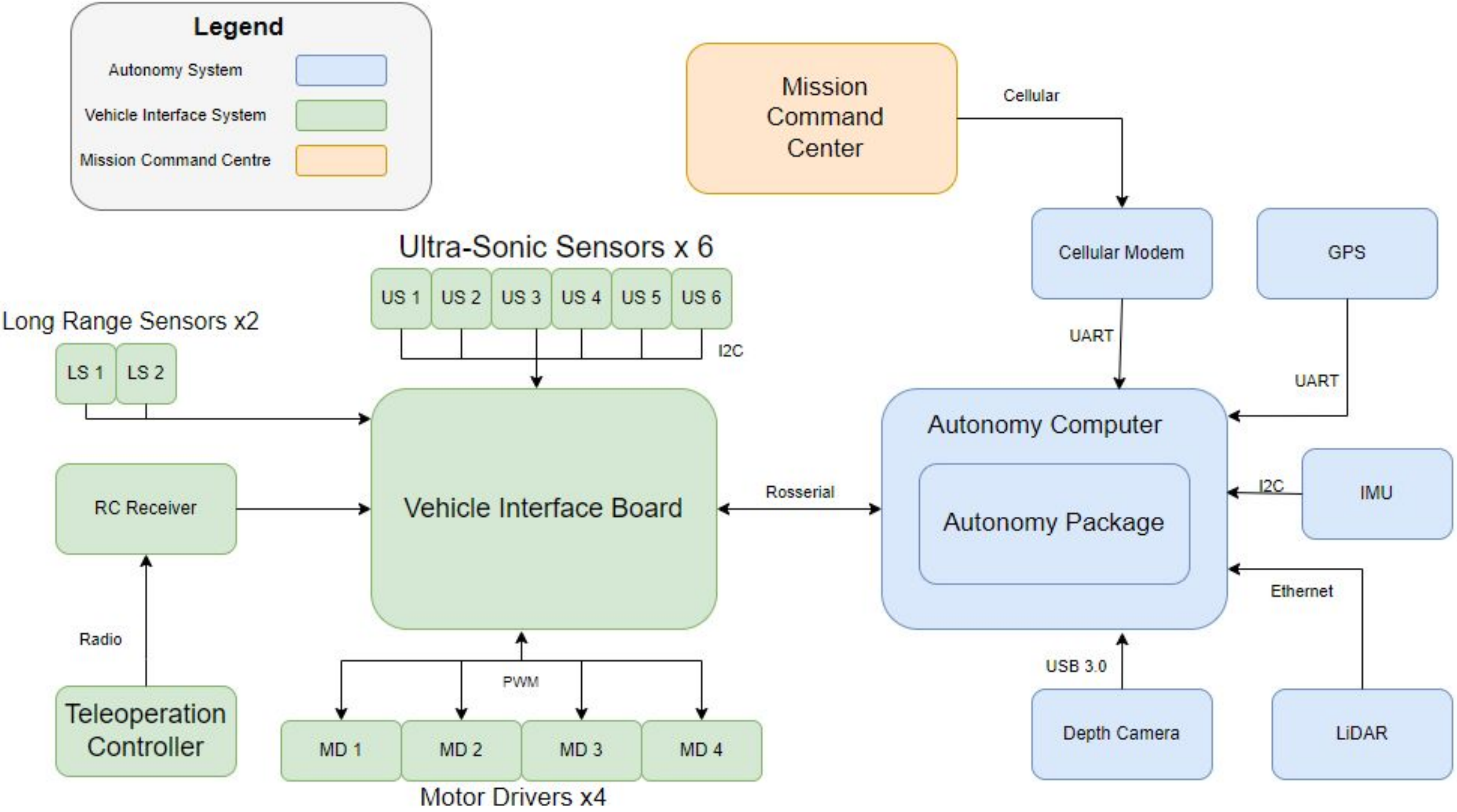
References

[1] hannahvsawiuk, “hannavsawiuk/Propbot” *GitHub*. [Online]. Available: https://github.com/hannahvsawiuk/Propbot/wiki/Software_Design. [Accessed: 29-Nov-2021].

[2] “Knowledge: Wheel control theory,” Robot Platform. [Online]. Available: http://www.robotplatform.com/knowledge/Classification_of_Robots/wheel_control_theory.html. [Accessed: 18-Apr-2022].

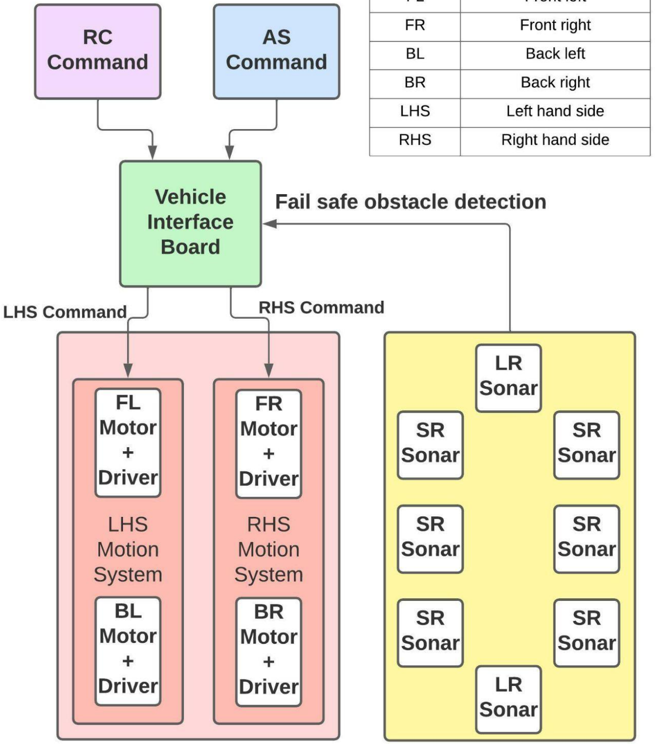
Appendix

Full System Overview



System Overview – Vehicle Interface

Legend	Definition
🟪	Remote Control
🟦	Autonomy System
🟩	Command Decision
🟨	Fail Safe Object Detection
🟥	Motor Control
LR	Long-range
SR	Short-range
FL	Front left
FR	Front right
BL	Back left
BR	Back right
LHS	Left hand side
RHS	Right hand side



RC Command

AS Command

Vehicle Interface Board

Fail safe obstacle detection

LHS Command

RHS Command

FL Motor + Driver

LHS Motion System

BL Motor + Driver

FR Motor + Driver

RHS Motion System

BR Motor + Driver

LR Sonar

SR Sonar

SR Sonar

SR Sonar

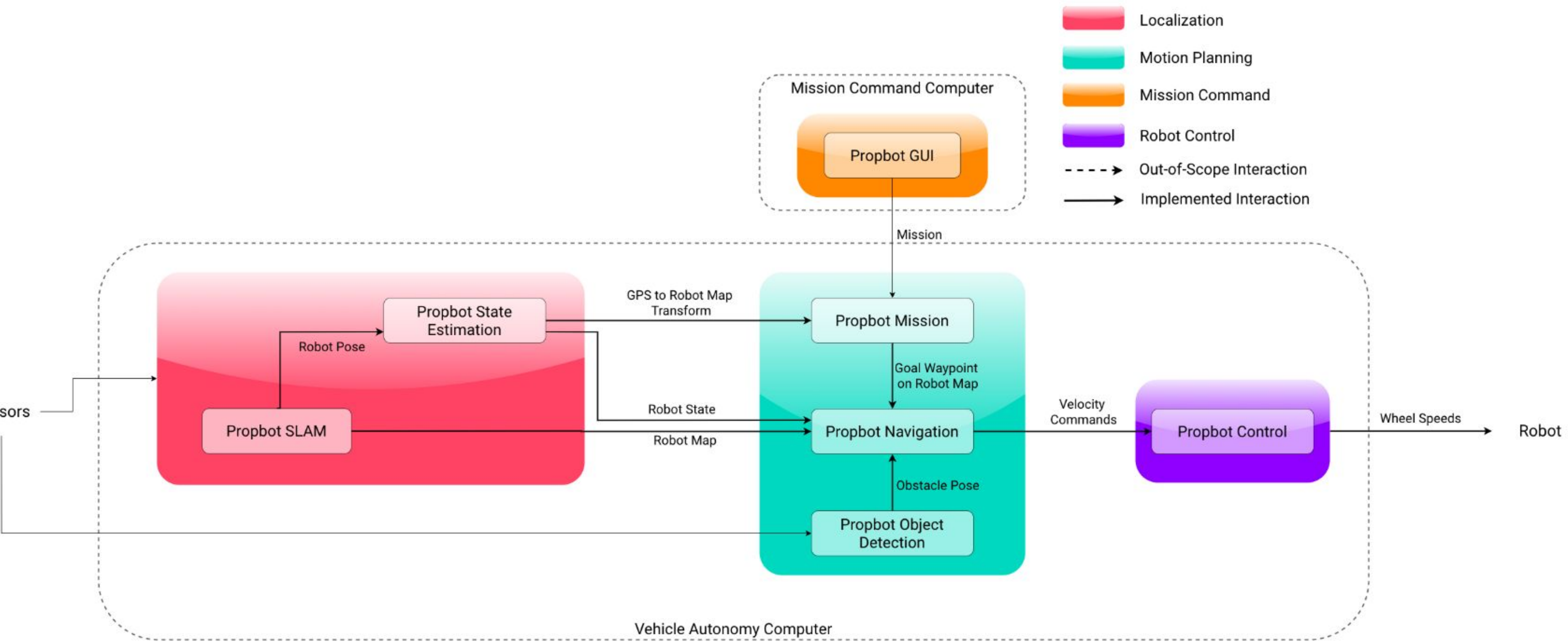
SR Sonar

SR Sonar

SR Sonar

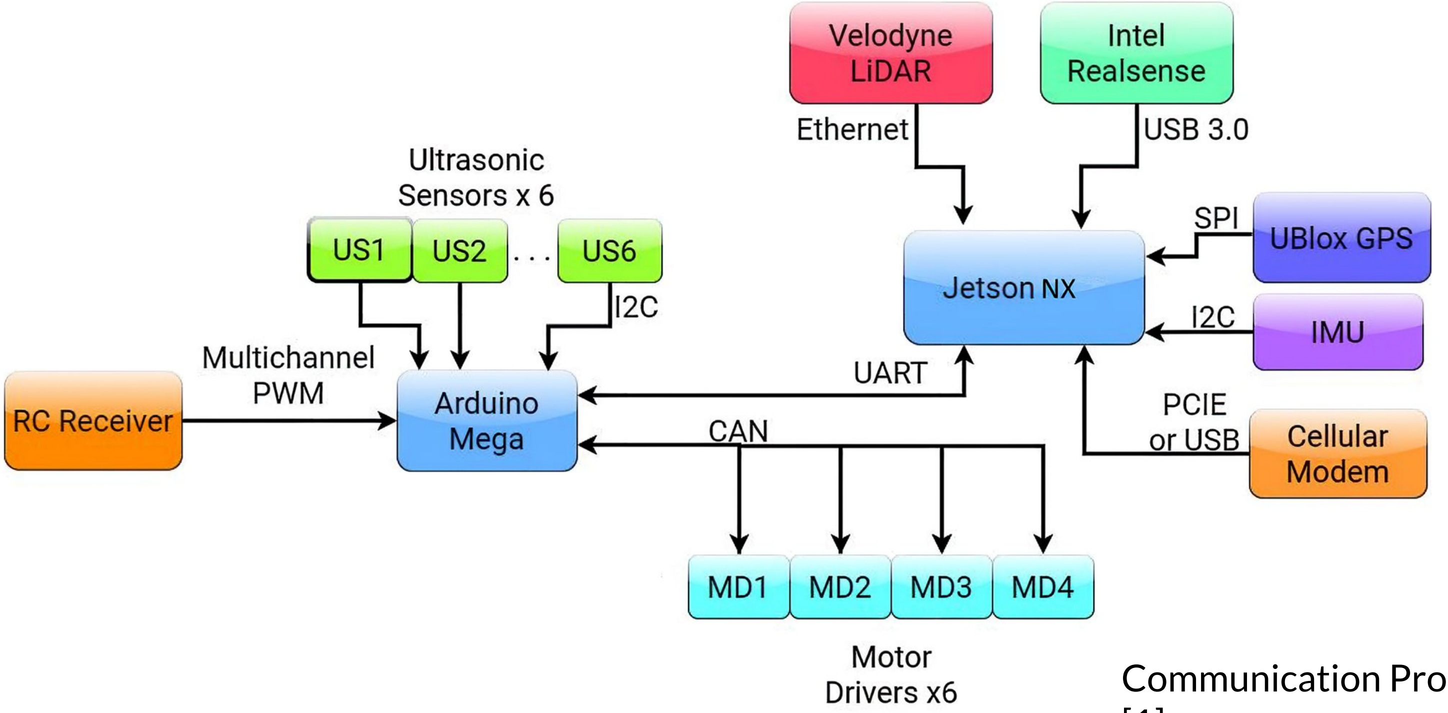
LR Sonar

System Overview – Vehicle Autonomy



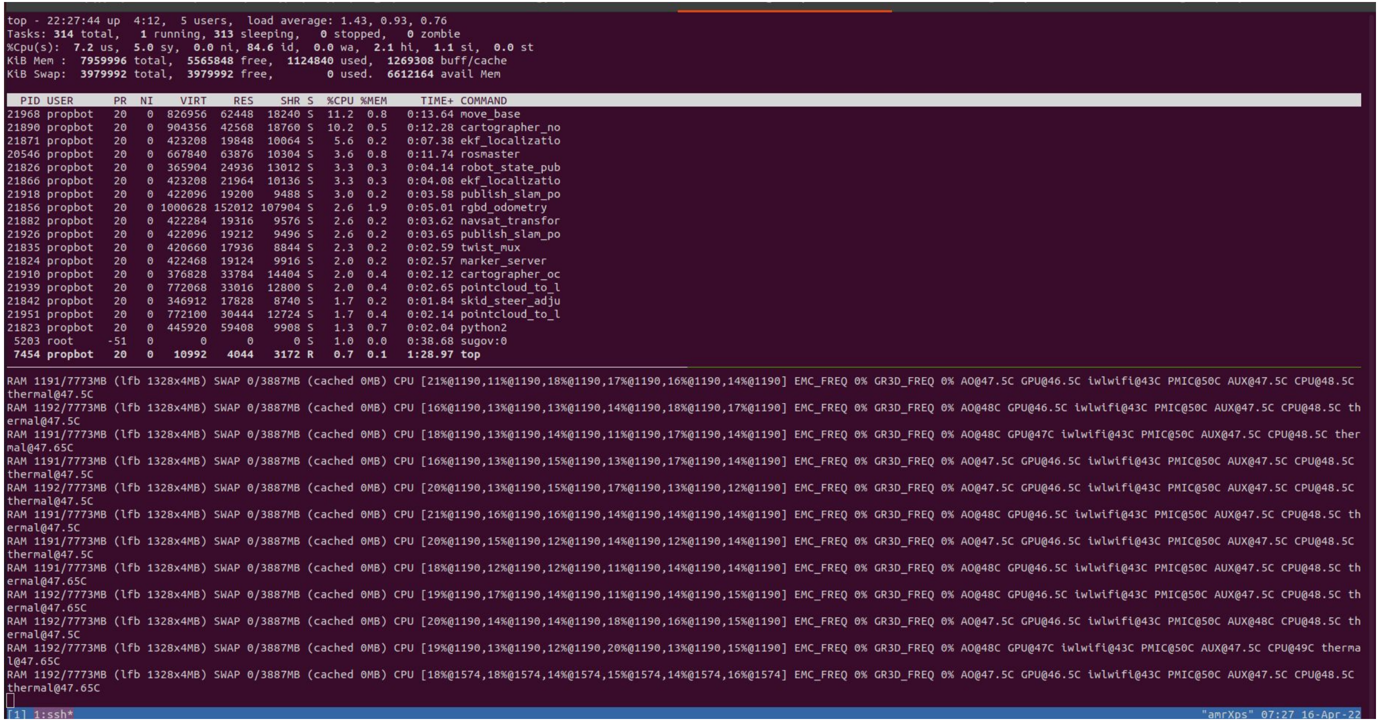
Autonomy Overview [1]

Communication Protocols



Communication Protocols [1]

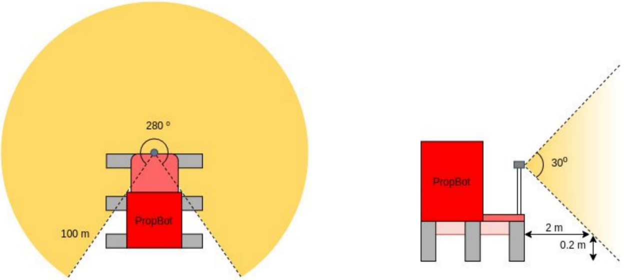
Autonomy Computer - Jetson NX



```
top - 22:27:44 up  4:12,  5 users,  load average: 1.43, 0.93, 0.76
Tasks: 314 total,   1 running, 313 sleeping,   0 stopped,   0 zombie
%Cpu(s):  7.2 us,  5.0 sy,  0.0 ni, 84.6 id,  0.0 wa,  2.1 hi,  1.1 si,  0.0 st
KiB Mem :  7959996 total,  5565848 free,  1124840 used,  1269308 buff/cache
KiB Swap:  3979992 total,  3979992 free,        0 used.  6612164 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
21968 propbot   20   0  826956  62448  18240 S  11.2  0.8   0:13.64 move_base
21890 propbot   20   0  904356  42568  18760 S  10.2  0.5   0:12.28 cartographer_no
21871 propbot   20   0  423208  19848  10064 S   5.6  0.2   0:07.38 ekf_localizatio
20546 propbot   20   0  667840  63876  10304 S   3.6  0.8   0:11.74 rosmaster
21826 propbot   20   0  365904  24936  13012 S   3.3  0.3   0:04.14 robot_state_pub
21866 propbot   20   0  423208  21964  10136 S   3.3  0.3   0:04.08 ekf_localizatio
21918 propbot   20   0  422096  19200   9488 S   3.0  0.2   0:03.58 publish_slam_po
21856 propbot   20   0 1000628 152012 107904 S   2.6  1.9   0:05.01 rgbd_odometry
21882 propbot   20   0  422284  19316   9576 S   2.6  0.2   0:03.62 navsat_transfor
21926 propbot   20   0  422096  19212   9496 S   2.6  0.2   0:03.65 publish_slam_po
21835 propbot   20   0  420660  17936   8844 S   2.3  0.2   0:02.59 twist_mux
21824 propbot   20   0  422468  19124   9916 S   2.0  0.2   0:02.57 marker_server
21910 propbot   20   0  376828  33784  14404 S   2.0  0.4   0:02.12 cartographer_oc
21939 propbot   20   0  772068  33016  12800 S   2.0  0.4   0:02.65 pointcloud_to_l
21842 propbot   20   0  346912  17828   8740 S   1.7  0.2   0:01.84 skid_steer_adju
21951 propbot   20   0  772100  30444  12724 S   1.7  0.4   0:02.14 pointcloud_to_l
21823 propbot   20   0  445920  59408   9908 S   1.3  0.7   0:02.04 python2
 5203 root     -51   0       0      0      0 S   1.0  0.0   0:38.68 sugov:0
 7454 propbot   20   0   10992   4044   3172 R   0.7  0.1   1:28.97 top

RAM 1191/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [21%@1190,11%@1190,18%@1190,17%@1190,16%@1190,14%@1190] EMC_FREQ 0% GR3D_FREQ 0% AO@47.5C GPU@46.5C iwlwifi@43C PMIC@50C AUX@47.5C CPU@48.5C
thermal@47.5C
RAM 1192/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [16%@1190,13%@1190,13%@1190,14%@1190,18%@1190,17%@1190] EMC_FREQ 0% GR3D_FREQ 0% AO@48C GPU@46.5C iwlwifi@43C PMIC@50C AUX@47.5C CPU@48.5C th
ermal@47.5C
RAM 1191/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [18%@1190,13%@1190,14%@1190,11%@1190,17%@1190,14%@1190] EMC_FREQ 0% GR3D_FREQ 0% AO@48C GPU@47C iwlwifi@43C PMIC@50C AUX@47.5C CPU@48.5C ther
mal@47.65C
RAM 1191/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [16%@1190,13%@1190,15%@1190,13%@1190,17%@1190,14%@1190] EMC_FREQ 0% GR3D_FREQ 0% AO@47.5C GPU@46.5C iwlwifi@43C PMIC@50C AUX@47.5C CPU@48.5C
thermal@47.5C
RAM 1192/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [20%@1190,13%@1190,15%@1190,17%@1190,13%@1190,12%@1190] EMC_FREQ 0% GR3D_FREQ 0% AO@47.5C GPU@46.5C iwlwifi@43C PMIC@50C AUX@47.5C CPU@48.5C
thermal@47.5C
RAM 1191/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [21%@1190,16%@1190,16%@1190,14%@1190,14%@1190,14%@1190] EMC_FREQ 0% GR3D_FREQ 0% AO@48C GPU@46.5C iwlwifi@43C PMIC@50C AUX@47.5C CPU@48.5C th
ermal@47.5C
RAM 1191/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [20%@1190,15%@1190,12%@1190,14%@1190,12%@1190,14%@1190] EMC_FREQ 0% GR3D_FREQ 0% AO@47.5C GPU@46.5C iwlwifi@43C PMIC@50C AUX@47.5C CPU@48.5C
thermal@47.5C
RAM 1191/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [18%@1190,12%@1190,12%@1190,11%@1190,14%@1190,14%@1190] EMC_FREQ 0% GR3D_FREQ 0% AO@48C GPU@46.5C iwlwifi@43C PMIC@50C AUX@47.5C CPU@48.5C th
ermal@47.65C
RAM 1192/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [19%@1190,17%@1190,14%@1190,11%@1190,14%@1190,15%@1190] EMC_FREQ 0% GR3D_FREQ 0% AO@48C GPU@46.5C iwlwifi@43C PMIC@50C AUX@47.5C CPU@48.5C th
ermal@47.65C
RAM 1192/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [20%@1190,14%@1190,14%@1190,18%@1190,16%@1190,15%@1190] EMC_FREQ 0% GR3D_FREQ 0% AO@47.5C GPU@46.5C iwlwifi@43C PMIC@50C AUX@48C CPU@48.5C th
ermal@47.5C
RAM 1192/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [19%@1190,13%@1190,12%@1190,20%@1190,13%@1190,15%@1190] EMC_FREQ 0% GR3D_FREQ 0% AO@48C GPU@47C iwlwifi@43C PMIC@50C AUX@47.5C CPU@49C therma
l@47.65C
RAM 1192/7773MB (lfb 1328x4MB) SWAP 0/3887MB (cached 0MB) CPU [18%@1574,18%@1574,14%@1574,15%@1574,14%@1574,16%@1574] EMC_FREQ 0% GR3D_FREQ 0% AO@47.5C GPU@46.5C iwlwifi@43C PMIC@50C AUX@47.5C CPU@48.5C
thermal@47.65C

[1] 1:ssh*                                                                                                     "amrXps" 07:27 16-Apr-22
```

LiDAR - Velodyne Puck VLP-16

- Provides 360 degree/100m range of high quality data (600000 points/sec) necessary for performing localization and mapping of Propbot’s environment
- VLP-16 is the most popular LiDAR used in mobile robotics applications and contains excellent software support
- Communicates and is powered via Ethernet

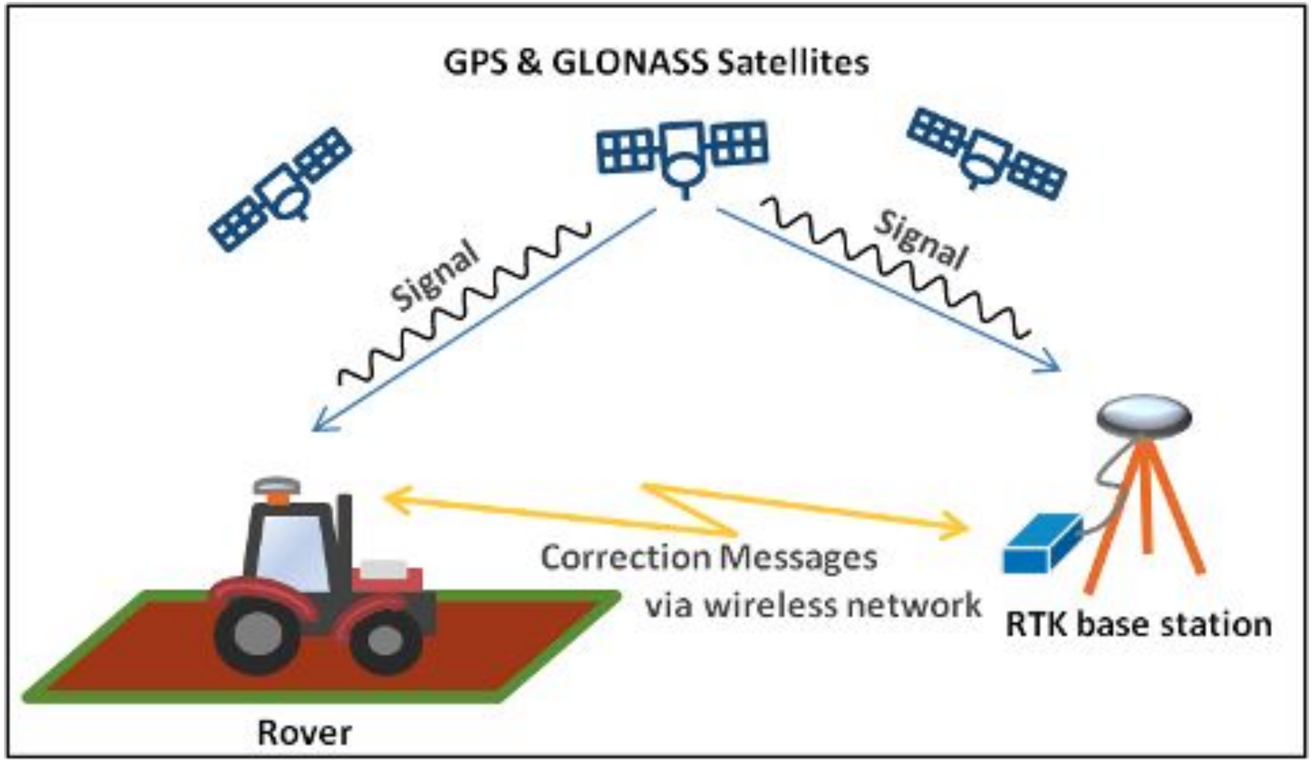


LiDAR Mapping [1]

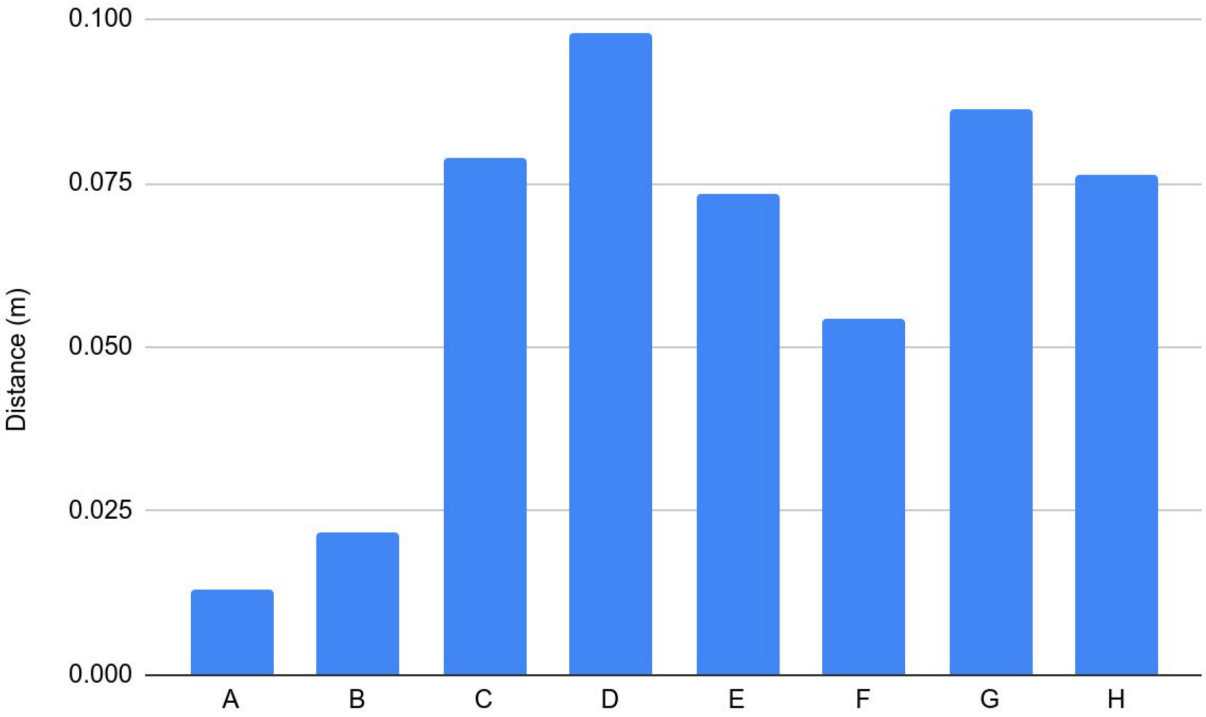
IMU - Bosch BNO055

- Provides orientation and acceleration vectors to the autonomy system
- Runs using existing ROS package based on an older version of the library, RTIMULib which is used widely in robotics community
- Calibration can be performed using the same library which stores calibration data in a file that is loaded whenever IMU is powered on

GPS - RTK System



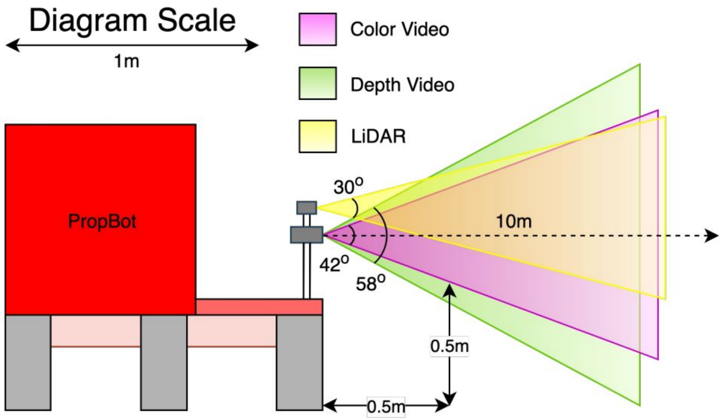
GPS - Validation

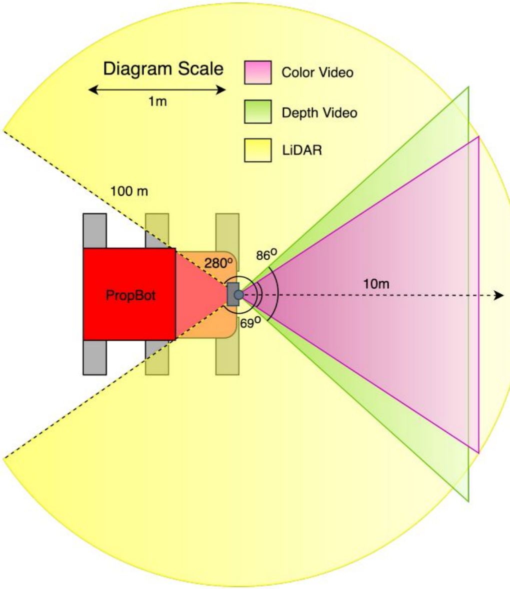


Depth Camera - Intel RealSense D435

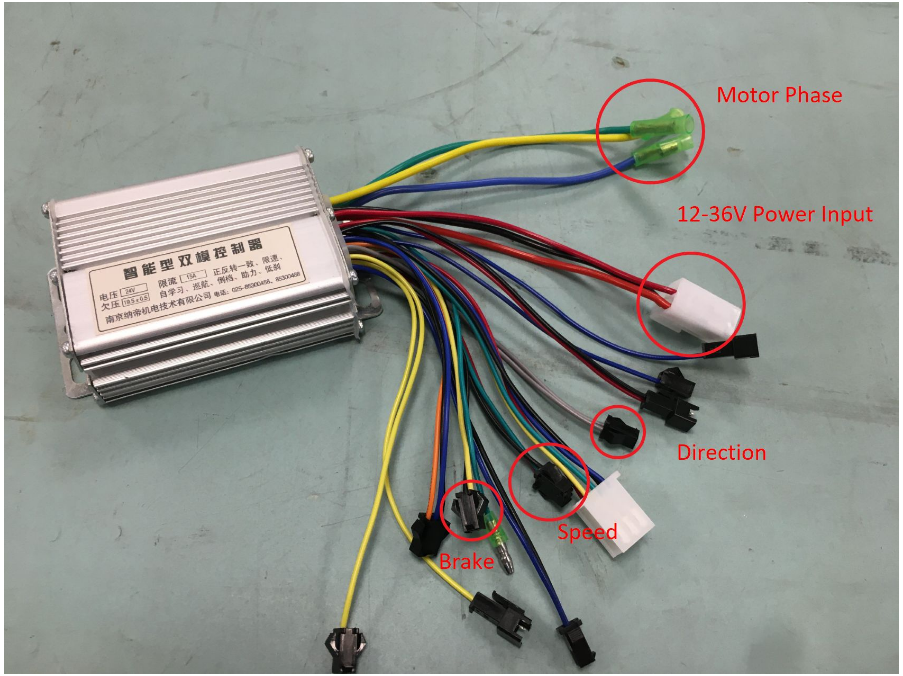
- 2 streams
    - Depth
        - 640 x 480 , 30 fps
    - Color
        - 640 x 480, 30 fps
- Validated by running visual odometry node

Sensor Mapping





Legacy Motor Controllers

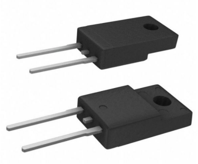


- Generic brushless DC motor controllers designed for use in DIY E-bike kits
- Variable analog voltage speed control
- Single-pin (Vcc/Gnd) direction and brake control

Roboteq Motor Controller Validation

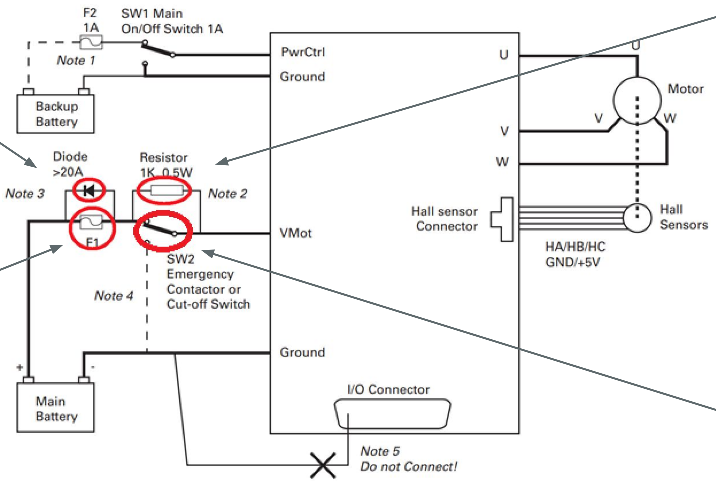
https://www.youtube.com/watch?v=buo4vMj-KM8&t=1s

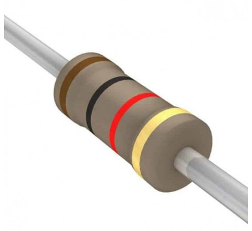
Roboteq Implementation Safety Features



Flyback Diode

Inline Fuse

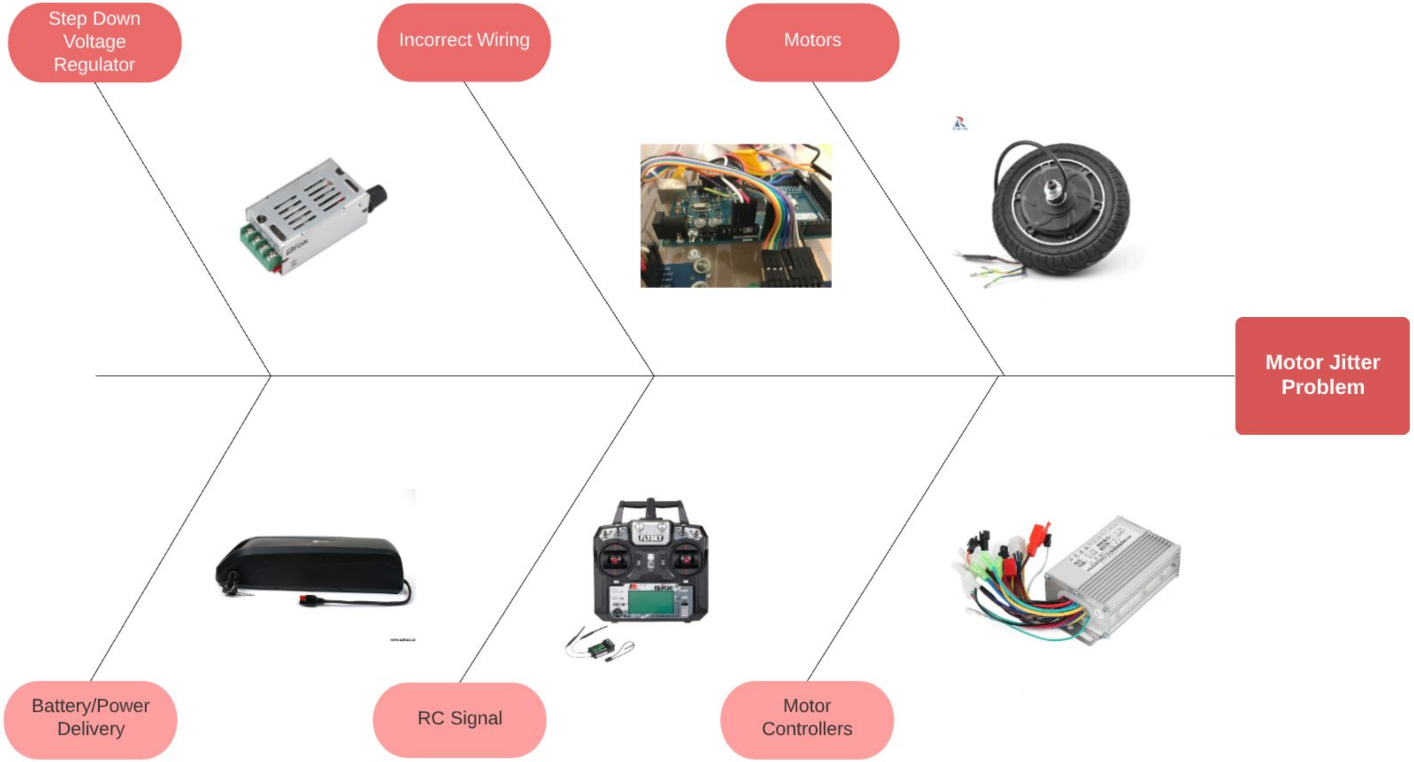




Precharge Resistor

E-Stop

Motor Jitter: Root Cause Analysis



Step Down Voltage Regulator

Incorrect Wiring

Motors

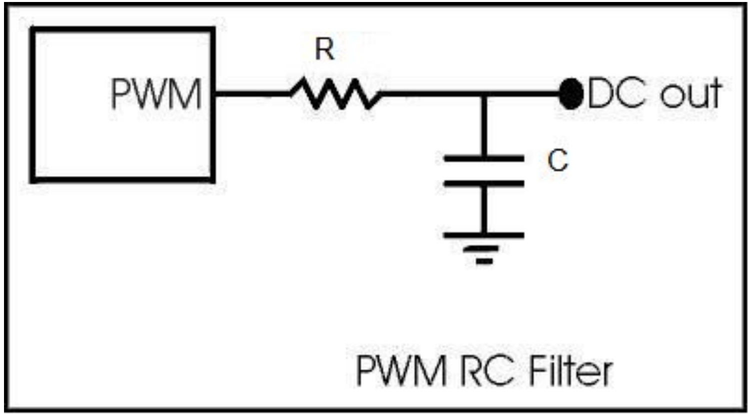
Motor Jitter Problem

Battery/Power Delivery

RC Signal

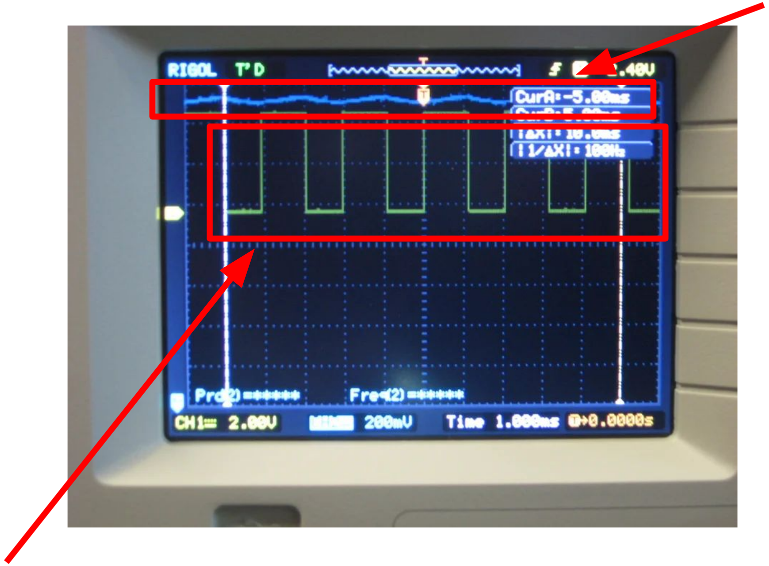
Motor Controllers

Legacy Motor Controller - RC Filter



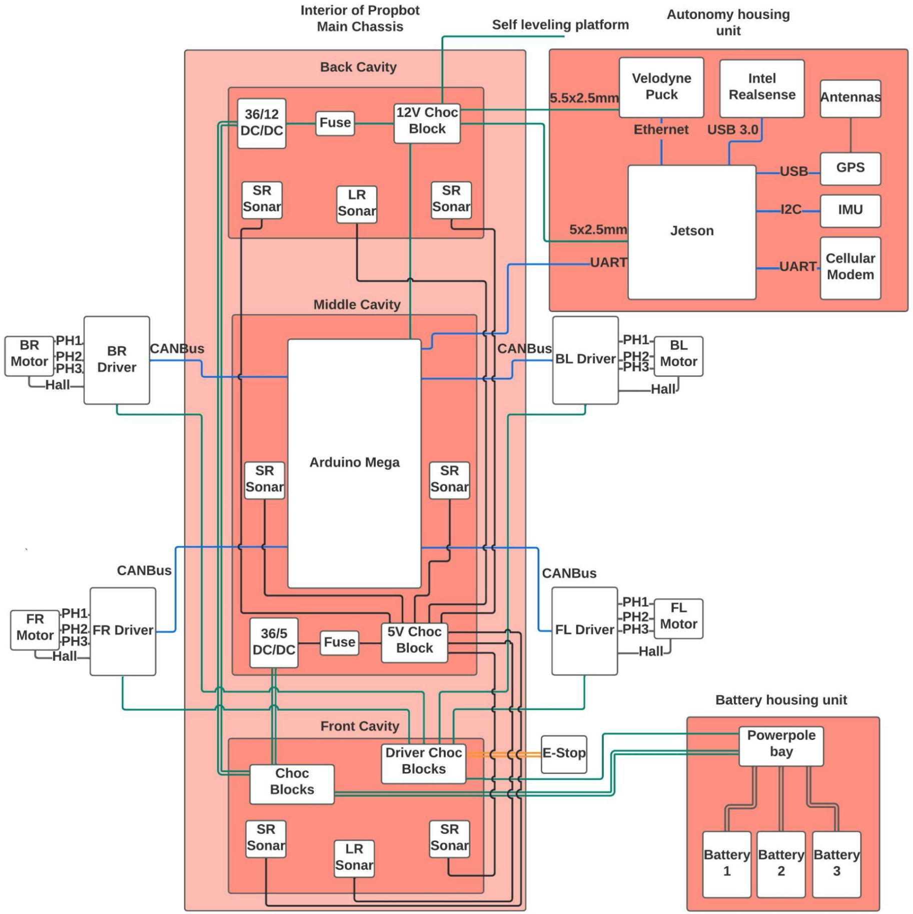
PWM RC Filter

Filtered output

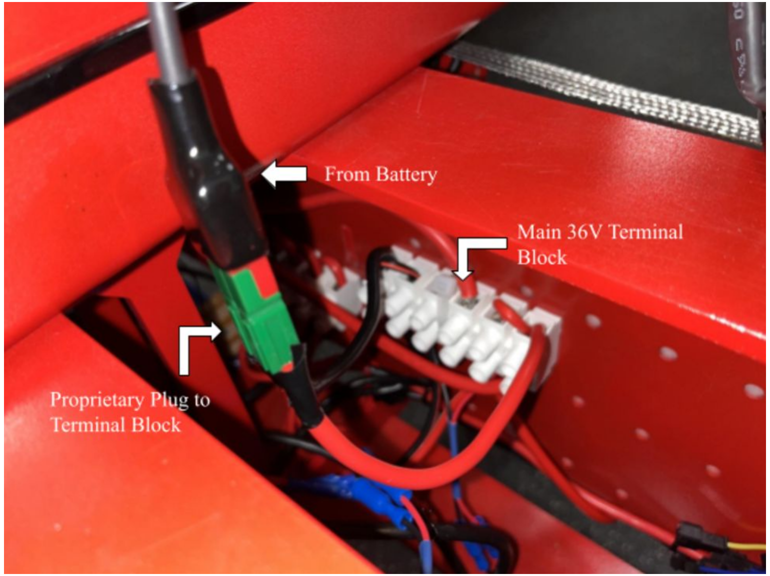


Unfiltered Arduino PWM Output

Power Conversion and Delivery



Power Conversion



From Battery

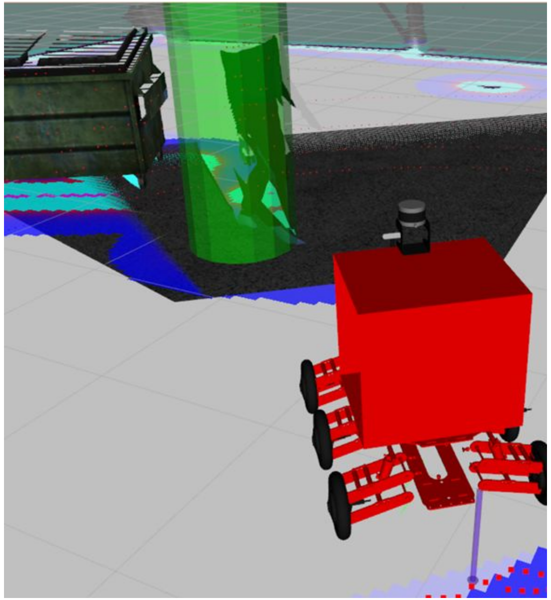
Main 36V Terminal Block

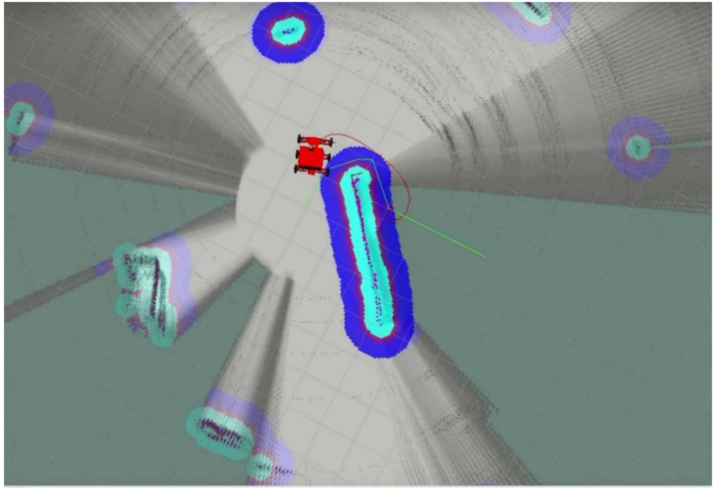
Proprietary Plug to Terminal Block

Autonomy System - Legacy

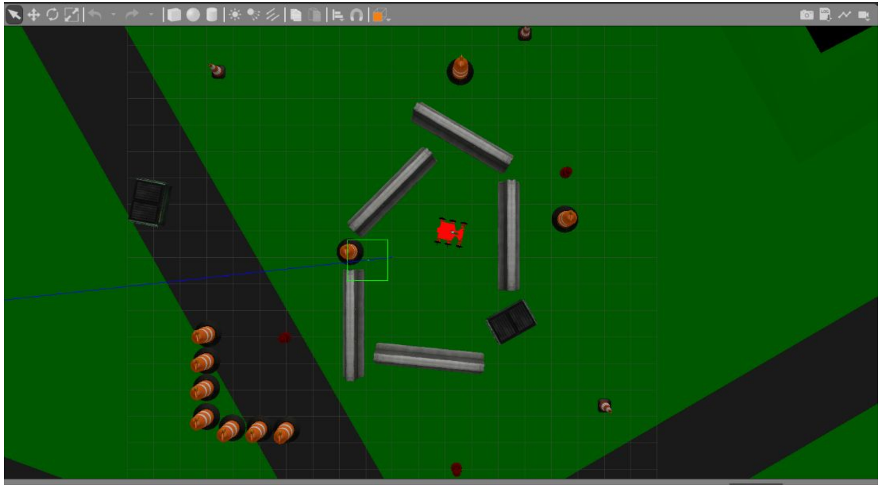
- Validation through simulation and visualization
- General Environment - not scenario specific
- Propbot control unrealistic

Simulation (ROS Gazebo)

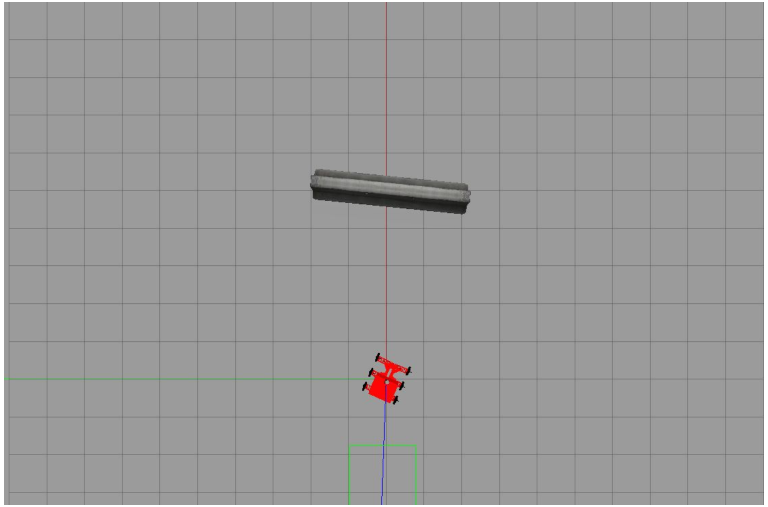




Autonomy Validation - Adding Scenarios

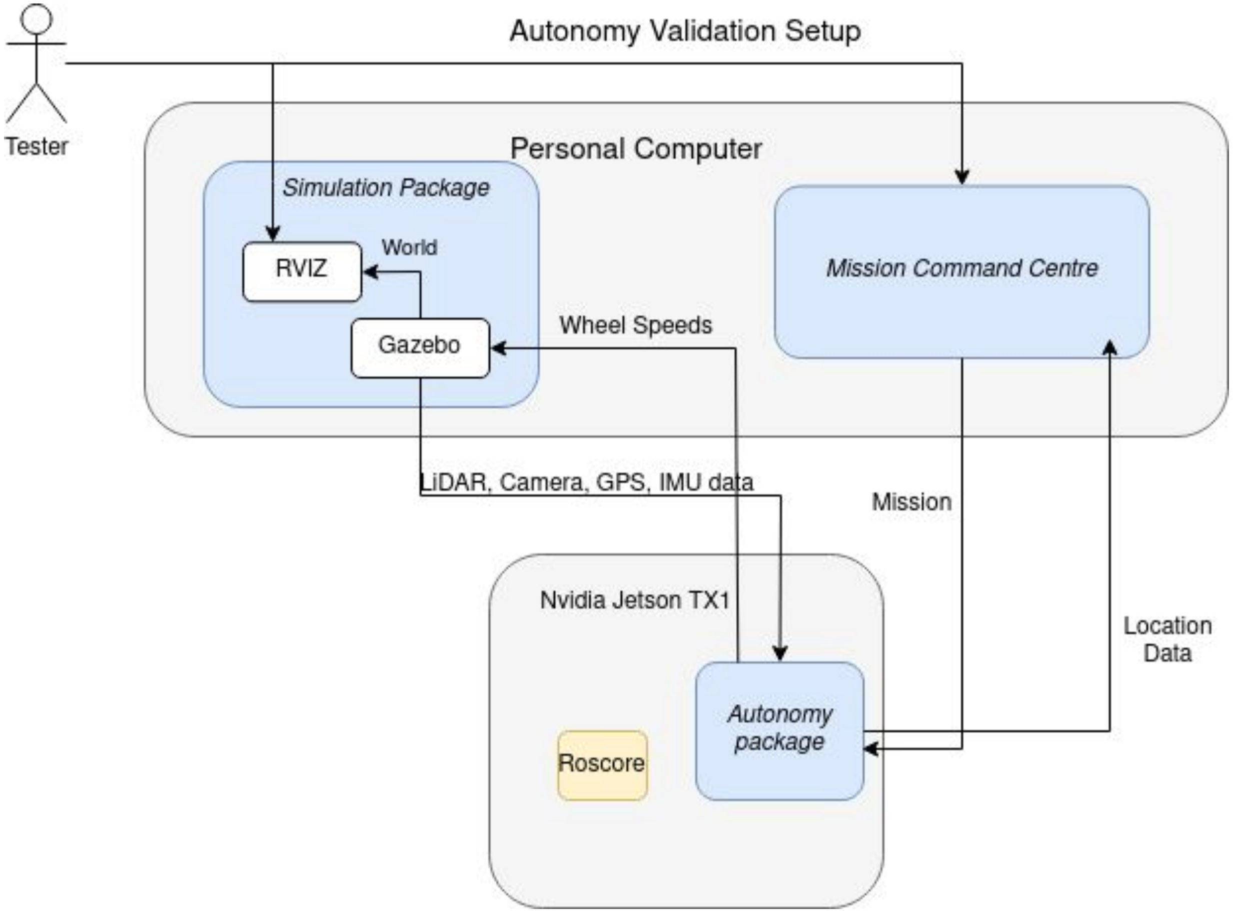


Situation specific simulations

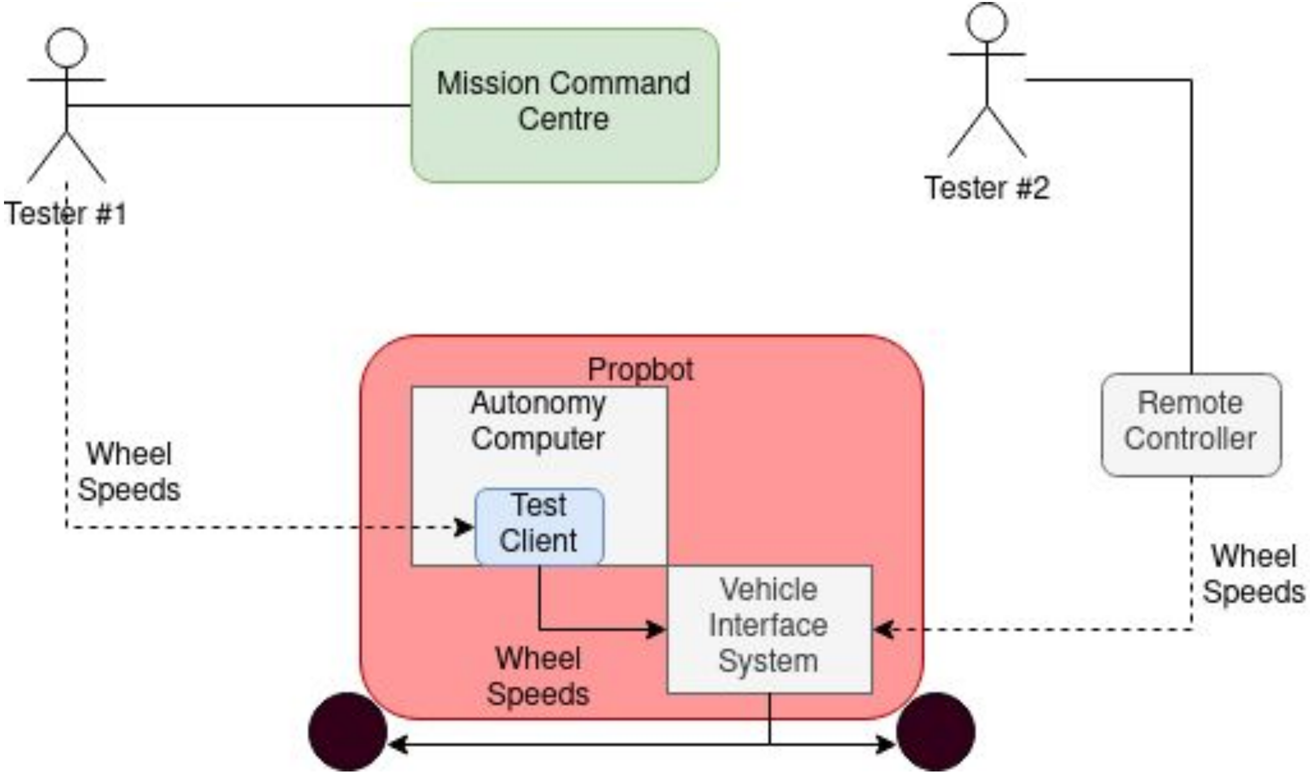


Resource-saving simulations

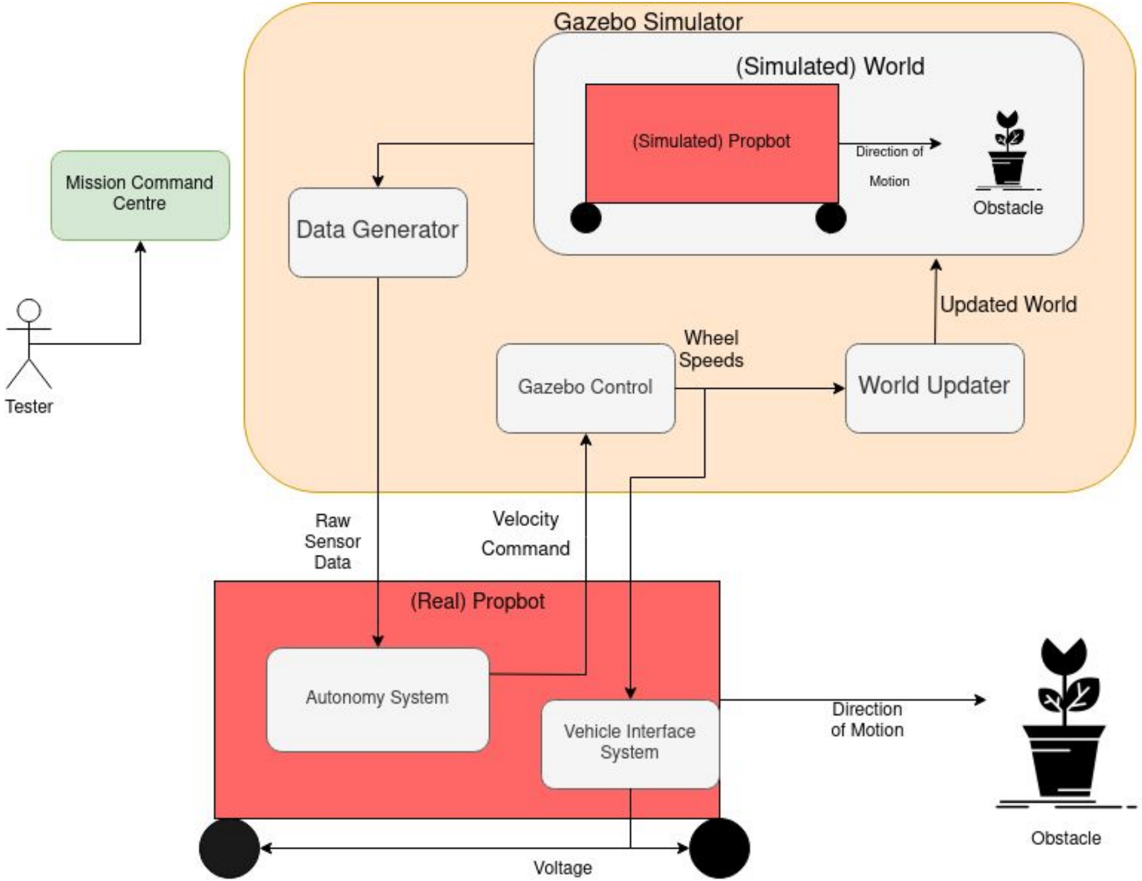
Autonomy Validation - Basic Setup



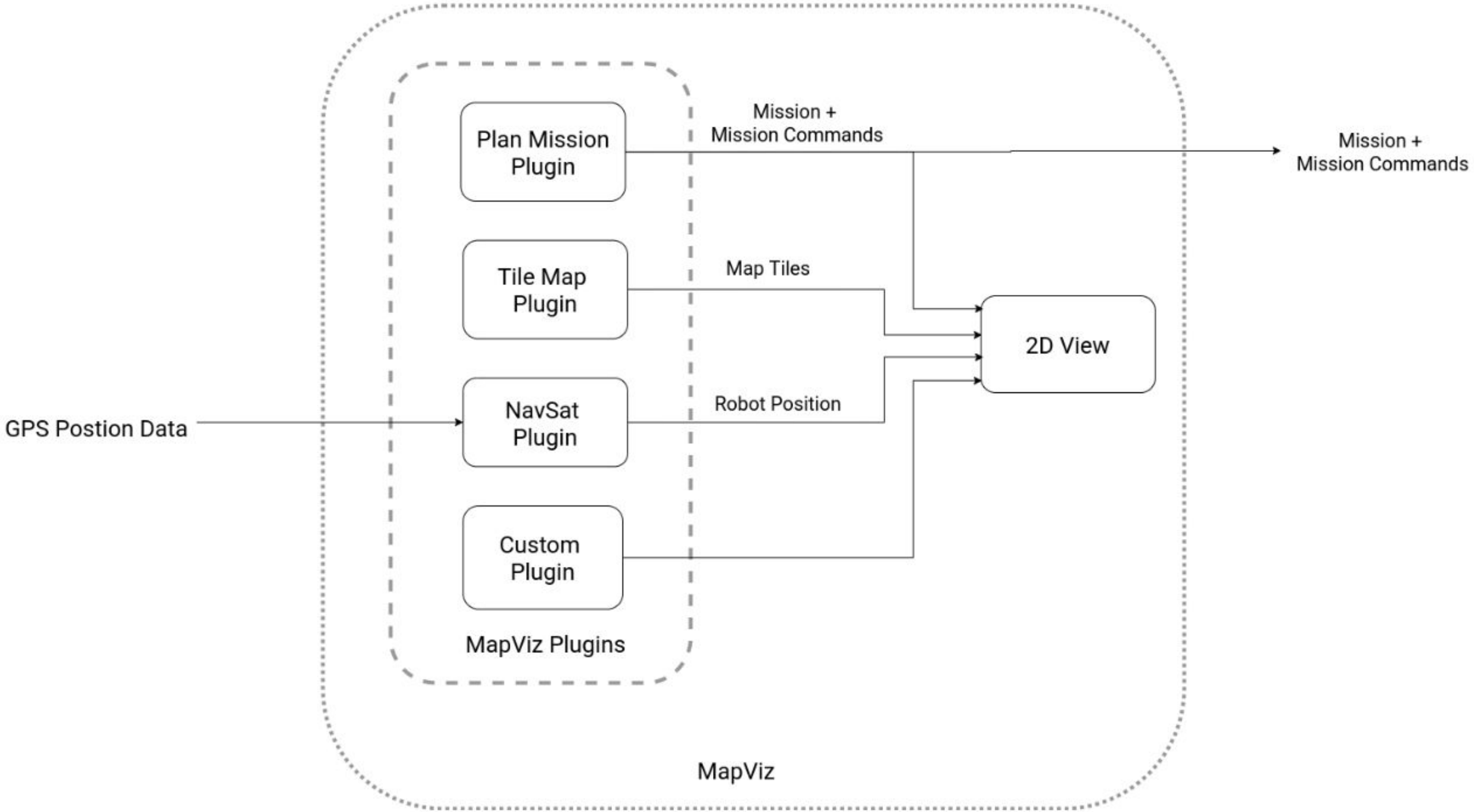
Validation Methods - Mocked Autonomy



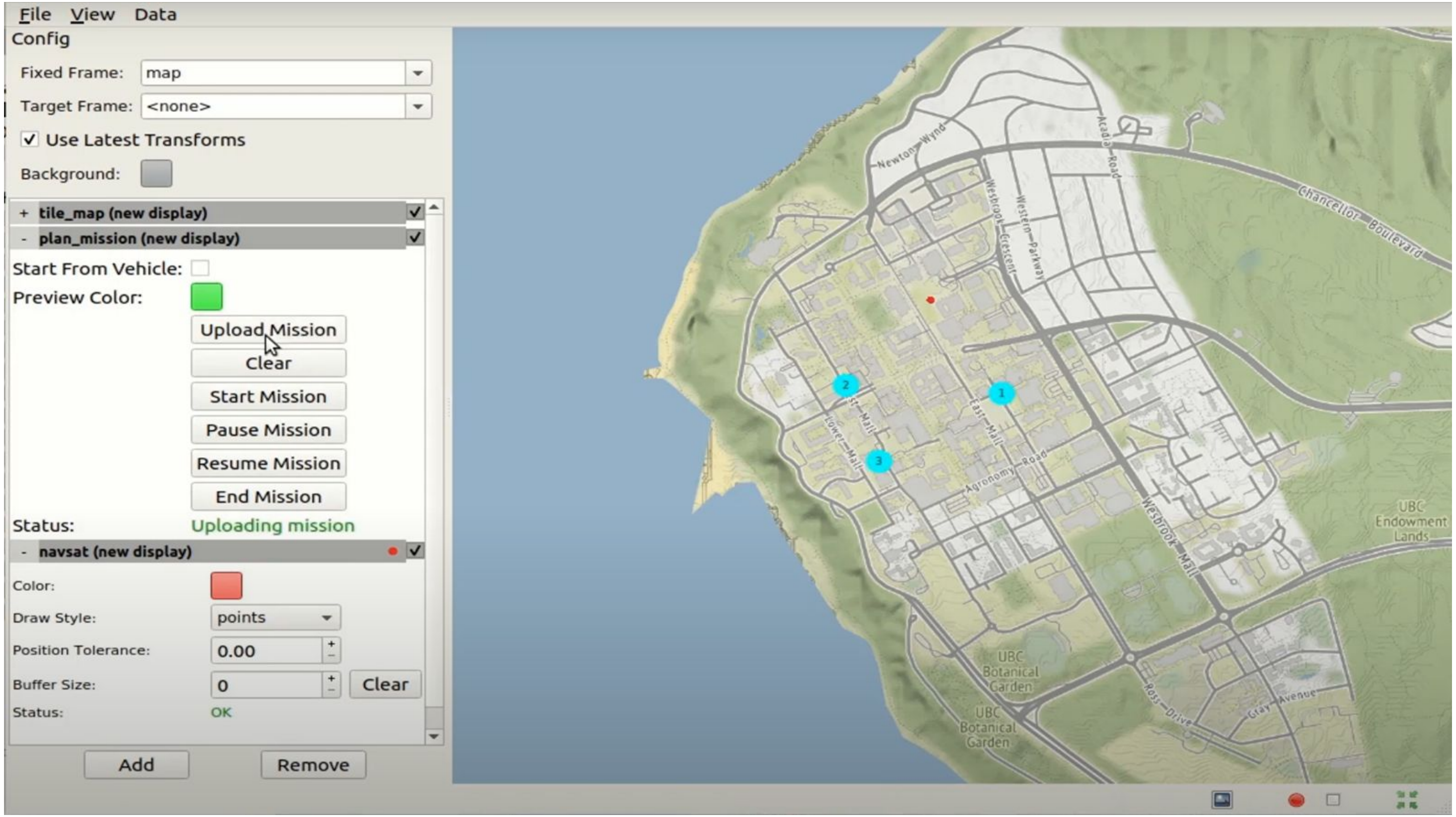
Validation Methods - Integrated Simulation



MCC (Mapviz) - Legacy Design



Mission Command Centre



Config

Fixed Frame: map

Target Frame: <none>

☑ Use Latest Transforms

Background:

+ tile_map (new display) ☑

- plan_mission (new display) ☑

Start From Vehicle: □

Preview Color: 🟩

Status: Uploading mission

- navsat (new display) ☑

Color: 🟥

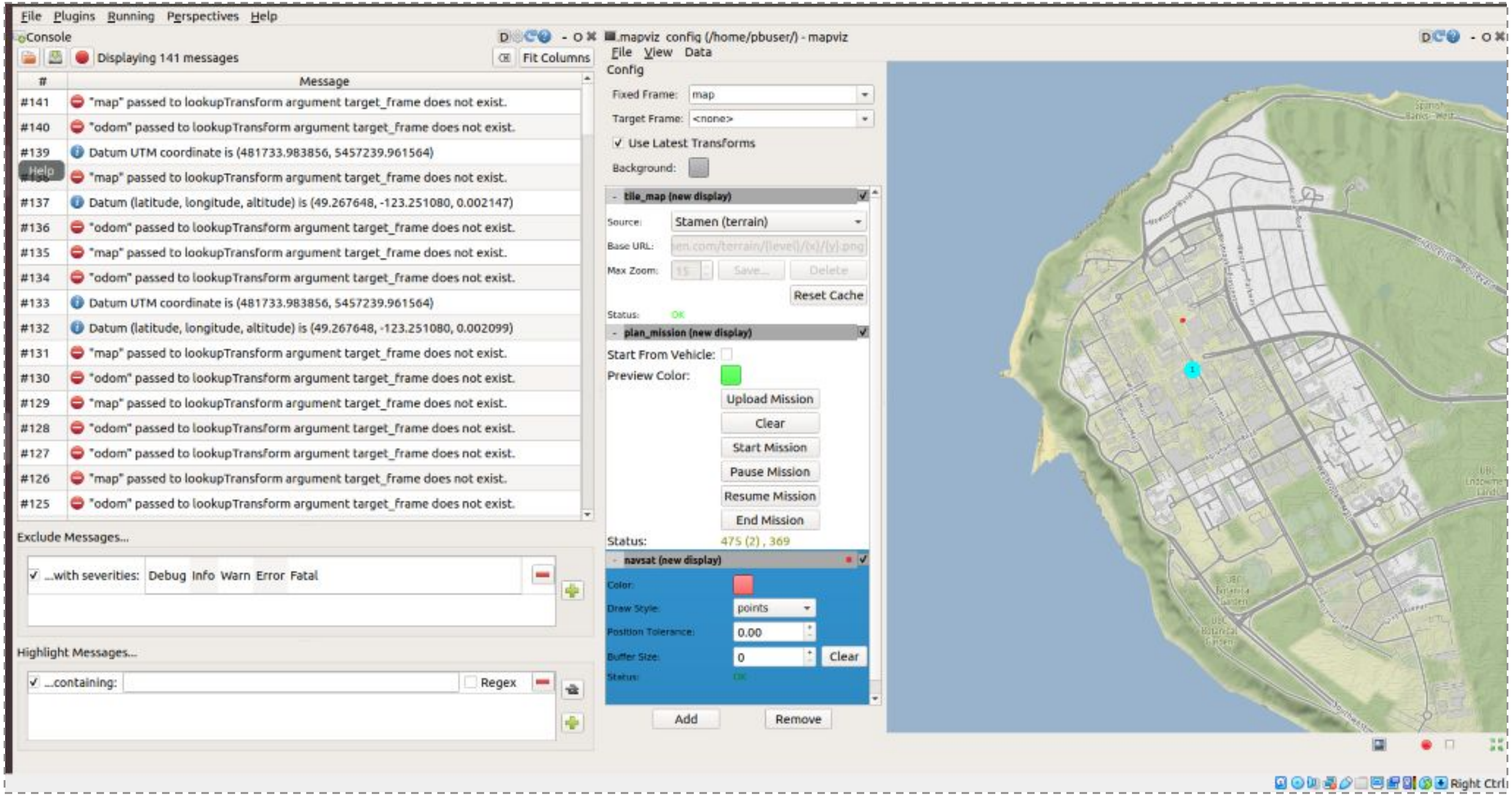
Draw Style: points

Position Tolerance: 0.00

Buffer Size: 0

Status: OK

Data Logging



Displaying 141 messages

```
#    Message
#141 "map" passed to lookupTransform argument target_frame does not exist.
#140 "odom" passed to lookupTransform argument target_frame does not exist.
#139 Datum UTM coordinate is (481733.983856, 5457239.961564)
#138 "map" passed to lookupTransform argument target_frame does not exist.
#137 Datum (latitude, longitude, altitude) is (49.267648, -123.251080, 0.002147)
#136 "odom" passed to lookupTransform argument target_frame does not exist.
#135 "map" passed to lookupTransform argument target_frame does not exist.
#134 "odom" passed to lookupTransform argument target_frame does not exist.
#133 Datum UTM coordinate is (481733.983856, 5457239.961564)
#132 Datum (latitude, longitude, altitude) is (49.267648, -123.251080, 0.002099)
#131 "map" passed to lookupTransform argument target_frame does not exist.
#130 "odom" passed to lookupTransform argument target_frame does not exist.
#129 "map" passed to lookupTransform argument target_frame does not exist.
#128 "odom" passed to lookupTransform argument target_frame does not exist.
#127 "odom" passed to lookupTransform argument target_frame does not exist.
#126 "map" passed to lookupTransform argument target_frame does not exist.
#125 "odom" passed to lookupTransform argument target_frame does not exist.
```

Exclude Messages...

✓ ...with severities: Debug Info Warn Error Fatal

Highlight Messages...

✓ ...containing: □ Regex

mapviz config (/home/pbuser/) - mapviz

Config

Fixed Frame: map

Target Frame: <none>

✓ Use Latest Transforms

Background:

tile_map (new display) ✓

Source: Stamen (terrain)

Base URL: ...en.com/terrain/{level}/{x}/{y}.png

Max Zoom: 15

Status: OK

plan_mission (new display) ✓

Start From Vehicle: □

Preview Color: 🟩

Status: 475 (2) , 369

navsat (new display) ✓

Color: 🟥

Draw Style: points

Position Tolerance: 0.00

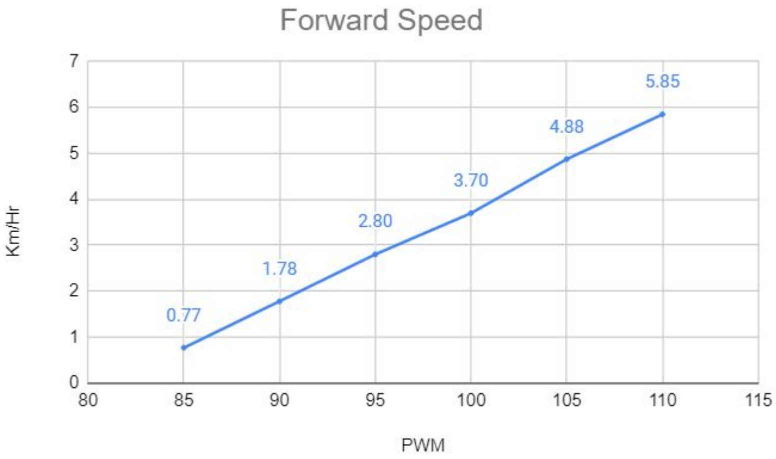
Buffer Size: 0

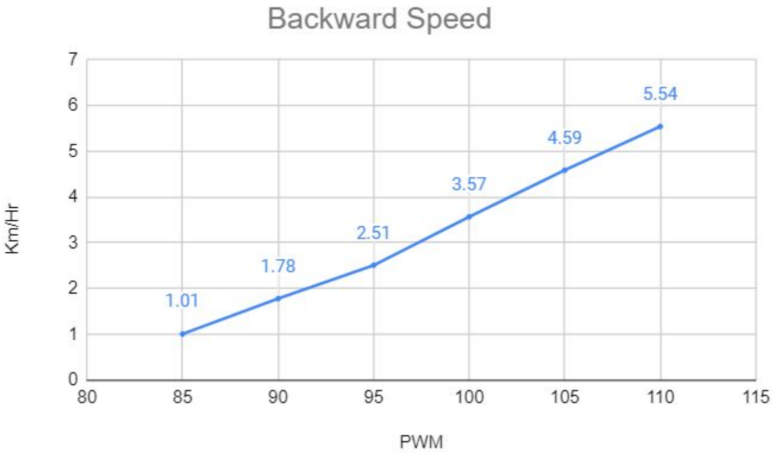
Status: OK

Data Collection

https://www.youtube.com/watch?v=jgco001iaqM

Speed Test Validation Results





Forward Max: 4.88km/hr
Forward Min: 1.78km/hr

Backward Max: 4.59km/hr
Backward Min: 1.01km/hr