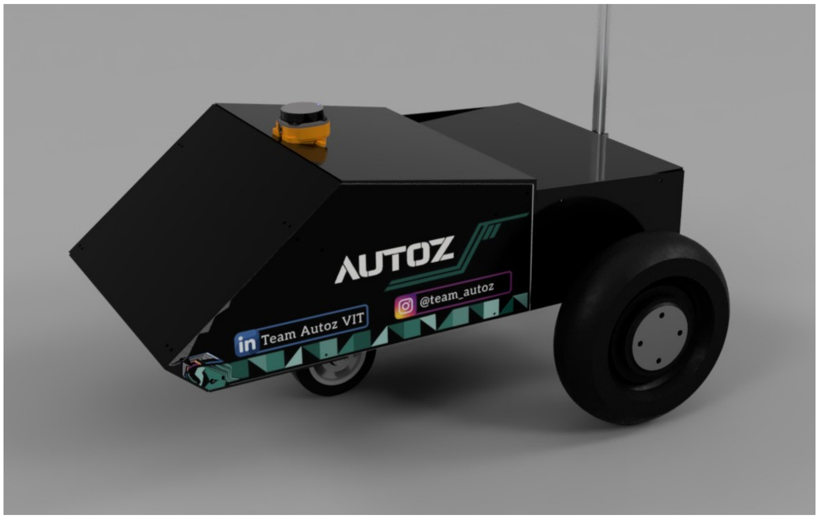
Team AutoZ - Qubi

Vellore Institute of Technology, Vellore

Intelligent Ground Vehicle Competition 2025



Date Submitted : 15 May, 2025

Faculty Advisors

Prof. S Denis Ashok | denisashok@vit.ac.in
Prof. Sathish Paulraj Gundupalli | sathish.gp@vit.ac.in

Team Lead

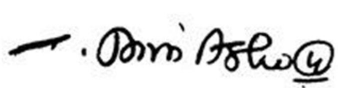
Rohan Jacob | rohan.jacob2022@vitstudent.ac.in

Members

Neha S A	nehasusan.anil2022@vitstudent.ac.in	Sushanth B	sushanth.bibi2024@vitstudent.ac.in
Vineeth P	vineeth.p2022@vitstudent.ac.in	Shreeshankar S	shreeshankar.s2024@vitstudent.ac.in
Kushaal K	kundala.kushaal2022@vitstudent.ac.in	Adarsh K S	adarsh.kumarsingh@vitstudent.ac.in
Paras J	paras.jain2022@vitstudent.ac.in	R Nikhil	nikhil.r2024@vitstudent.ac.in
Abir P	abir.pathania2022@vitstudent.ac.in	Srinidhi R	srinidhi.roy2024@vitstudent.ac.in
Basil S V	basil.shaiju2022@vitstudent.ac.in	Riddhi B	riddhi.bhaduri2024@vitstudent.ac.in
Aayush N	aayush.nambiar2023@vitstudent.ac.in	Shynaa C	shynaa.chopra2024@vitstudent.ac.in
Palak S	palak.saxena2023@vitstudent.ac.in	Chhavi S	chhavi.sekwal2023@vitstudent.ac.in
George V K	george.varghese2024@vitstudent.ac.in	Bonala M	bonala.mansi2024@vitstudent.ac.in
Elamaran Y	elamaran.y2024@vitstudent.ac.in	Aahana N	aahana.nath2024@vitstudent.ac.in

Statement of Integrity

I hereby certify that the development of the bot, Qubi, as described in this report, is equivalent to the work involved in a senior design course. This report has been prepared by the students of Team AutoZ under my guidance.



Prof. Denis Ashok Sathiaseelan, Faculty Advisor

Contents

1 Introduction and Team Overview

1.1 Intoduction

Team AutoZ is a group of interdisciplinary undergraduates from VIT Vellore, specializing in autonomous technologies across a wide spectrum—from Unmanned Aerial Vehicles (UAVs) to Unmanned Ground Vehicles (UGVs). In addition to our research and development efforts in robotics, we actively provide industrial automation solutions through consultation-based projects. For this year's competition, we present our bot Qubi, a newer improved version of Pratham, our bot that we used for IGVC 2023. Qubi has been upgraded in terms of hardware and software to meet the current challenge requirements.

1.2 Team Organization

The board of 2024-25:

Role	Name	Domain	Major
Team Lead	Rohan Jacob	3rd Year	Electronics
Head of Research and Development	Neha Susan Anil	3rd Year	Electronics
Head of Embedded Systems	Vineeth Prashanth	3rd Year	Electronics
Head of Electrical Hardware Systems	Kushaal Kundala	3rd Year	Electrical
Head of Manufacturing	Paras Jain	3rd Year	Mechanical
Head of Mechanical Systems	Abir Pathania	3rd Year	Mechanical
Head of Management	Basil Shaiju Velakkattukuzhi	3rd Year	Computer Sc.

The juniors:

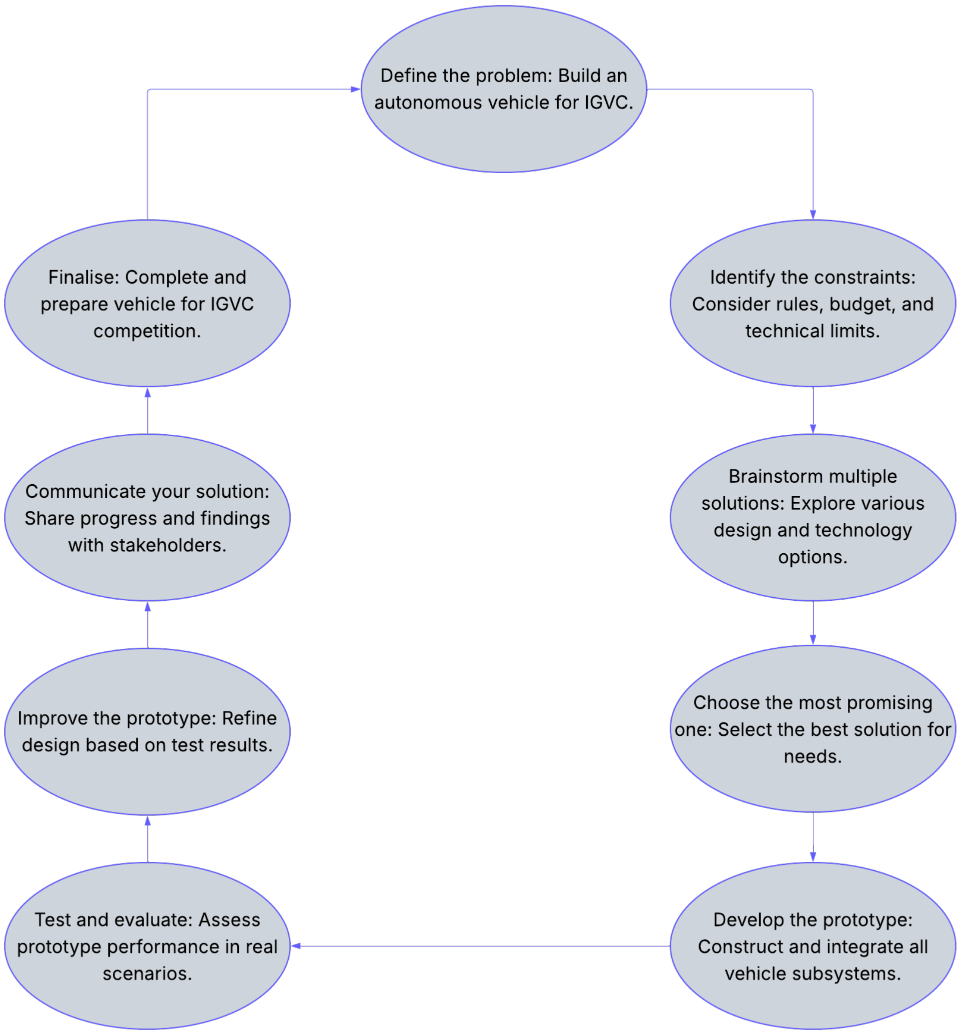
Domain	Name	Year	Major
Mechanical	Aayush Nambiar	2nd Year	Mechanical
	Palak Saxena	2nd Year	Mechanical
	George Varghese Kurian	1st Year	Mechanical
	Shreeshankar S	1st Year	Mechanical
	Elamaran Y	1st Year	Mechanical
	Sushanth Bibi	1st Year	Mechanical
Electrical & Electronics	Adarsh Kumar Singh	1st Year	Electrical
	R Nikhil	1st Year	Electronics
	Srinidhi Roy	1st Year	Electronics
Autonomy	Riddhi Bhaduri	1st Year	Computer Sc.
	Shynaa Chopra	1st Year	Computer Sc.
Management	Chhavi Sekwal	2nd Year	Computer Science
	Bonala Mansi	1st Year	Mechanical
	Aahana Nath	1st Year	Mechanical

1.3 Design Assumptions and Process

Our team followed a sequential engineering method to design our autonomous ground vehicle for the IGVC competition. We started by clearly defining the problem according to the IGVC rules and requirements. After that, we conducted research into current technologies and solutions relevant to autonomous ground vehicles. Using this information, we brainstormed a range of ideas and then assessed them to select the most feasible options. These ideas were organized into specific subsystems, covering mechanical, electrical, and software aspects.

We then assembled and integrated these subsystems, regularly testing our vehicle in both simulations and real-world conditions. Each round of testing informed improvements and refinements to our design.

Throughout the process, we maintained open communication and detailed documentation to keep our workflow synchronized and to ensure there was no ambiguity. This sequential engineering method helped us systematically address each challenge and continuously enhance our autonomous ground vehicle.

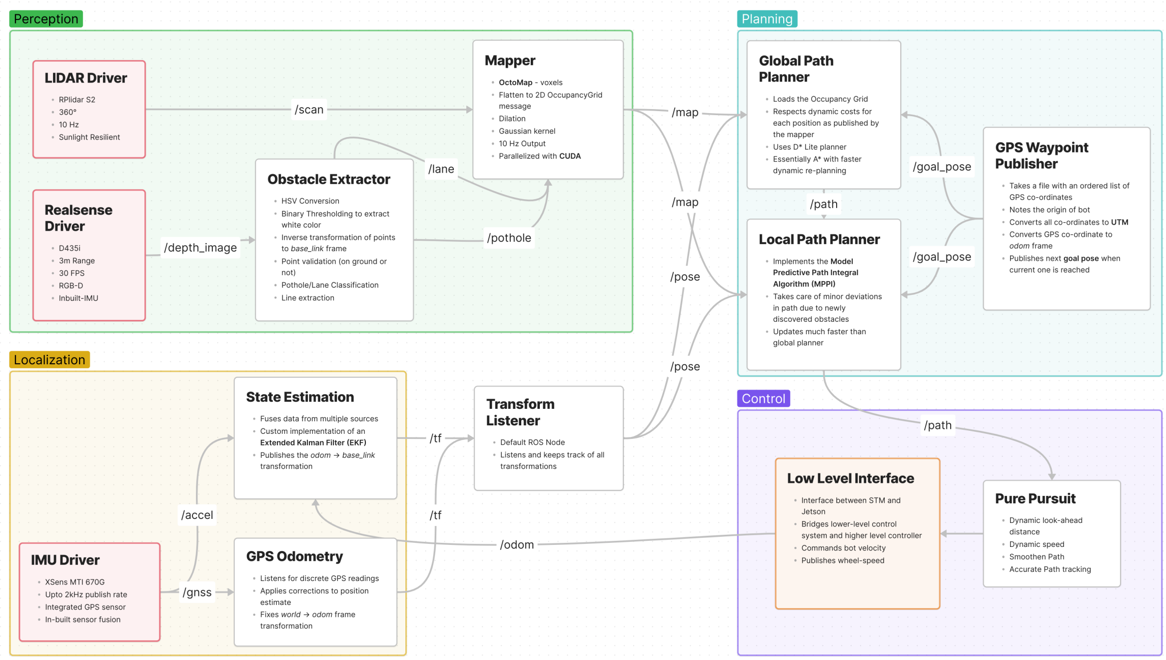


2 System Architecture

2.1 Safety Devices

The main MCB is placed to automatically interrupt the electrical circuit during overloads or short circuits, preventing damage to components and wiring, and a fuse is placed on the power distribution boards to prevent damage of circuitry in case the input current is too high. Wireless and wired E-stops are present on the bot to manually cut off current supply to the motors in case of emergencies, when the bot behaves unpredictably. There is a diagnostics board which consists of current sensors which detects anomalies in the current usage and cuts off supply when required.

2.2 Software Modules



2.3 Components & Cost Breakdown

Item	MSRP (in USD, Total)	Quantity	Cost to Team (in USD)
Xsens MTI-670G (IMU+GNSS)	3090	1	1545
RP-LIDAR-S2	395	1	0
Realsense-D415	450	1	0
Motor-Drive (MDDS30)	75	1	0
Motors–Brushed	135	2	0
PCBs	495	3	235
Estop-Boards	25	2	25
MCB	8	1	8
Wires,Connectors	20	1	20
Mild Steel	15	2	15
Outer-Casing Acrylic	75	1	75
3D Prints	50	12	50
Motor-Shafts, Brackets	20	2	20
Wheels	80	2	80
Miscellaneous (Nuts,Bolts,etc)	20	1	20
Total Cost			**2093**

3 Effective Innovations

- **Welded Mild Steel Frame:** For the chassis of Qubi, we selected mild steel (MS) square bars over aluminium extrusion pipes due to their higher tensile strength, better impact resistance, and lower fabrication cost. This choice allowed us to build a durable and cost-effective frame without compromising on structural integrity. We used MIG welding to join the MS bars, enabling us to fabricate the entire chassis as a single, solid piece. This eliminated the need for nuts and bolts, which not only simplified the assembly process but also significantly reduced mechanical vibrations. As a result, there is less risk of components loosening over time, and the performance of onboard electronic sensors is improved due to reduced vibration-related interference.
- **RealSense Camera Mount with Active Cooling System:** We custom-designed and 3D-printed the camera mount using PLA material, incorporating built-in grooves and mounted with side-blowing fans to enable active cooling. This innovative design helps prevent the RealSense camera from overheating by maintaining optimal operating temperatures during use. As a result, we can ensure consistent camera performance and reliability, even during extended periods of operation.
- **Modular Drive Train:** Our modular drive train is designed to offer flexibility and adaptability, allowing for an easy transition between 2-wheel drive (2WD) and 4-wheel drive (4WD) setups. This feature enables the system to respond effectively to different terrain types and torque demands. The switch requires only minimal changes like adding motors and making slight adjustments to the control code. This makes it both efficient and user-friendly. This design not only improves performance but also simplifies maintenance and future upgrades. The chassis already includes dedicated areas for motor mounts, so adding a motor and wheel to upgrade from 2WD to 4WD is straightforward. The modular approach also supports rapid prototyping and testing, making it ideal for research, development, and real-world applications where versatility is essential.
- **Current Sensor Safety:** We implemented additional safety feature on bot using WCS1700 hall effect based current sensor to measure and detect any abnormal usage of current by the motors and cut the power to it if detected.

4 Mechanical Design

4.1 Overview

Building on the lessons learned from our previous robot, Pratham (2023), we identified and addressed its shortcomings to deliver significant improvements in the new model, Qubi. Key enhancements include optimized dimensions for better compliance and manoeuvrability, a redesigned drive train for improved reliability, and effective measures to reduce vibration and prevent loosening of nuts and bolts. Throughout the development process, we relied on Fusion 360 for detailed CAD modelling and ANSYS for structural simulations, ensuring all components met stringent strength and durability standards. The manufacturing process incorporated a range of advanced techniques, such as laser cutting, FDM 3D printing, MIG welding, and precision machining. These combined efforts have resulted in a more dependable, maintainable, and competition-ready robot.

4.2 Structural Designs

- **Materials and Joining Techniques:**
  One of the key problems we encountered in Pratham was related to the use of T-slot aluminium extrusion bars. Over time, the L-clamps used to join these bars would loosen due to vibrations during operation. This not only affected the overall structural integrity of the robot but also introduced noise into the sensor readings, leading to less reliable performance. To solve this, we switched to using 20x20 mm mild steel (MS) square bars and welded them together to form a single, solid chassis. This made the frame much more rigid and durable. Welding eliminated the need for joints that could loosen, resulting in a significantly stronger and vibration-resistant structure. We have also used a 20x20 T-slot

aluminium extrusion bar as a pole at the back of the chassis to mount a Realsense camera on it. We have used rubber dampers to reduce the vibration in the pole.

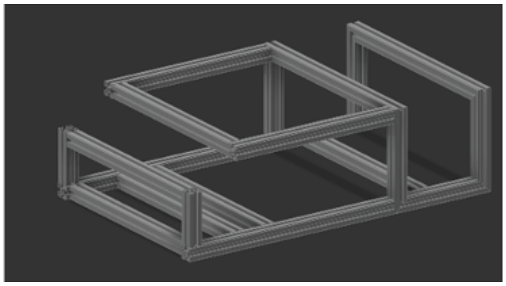


Figure 1: Pratham Chassis (2023)

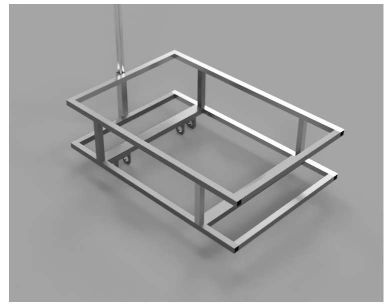


Figure 2: Qubi Chassis (2025)

- **Outer Casing and Platform:**
  We chose acrylic sheets for the outer casing and base platform. Acrylic is lightweight, easy to fabricate and provides a sleek, professional finish. It was also ideal for our design needs because it has a good balance between strength and weight. Using acrylic helped reduce the overall weight of the robot, making it more agile and efficient, while providing good weatherproofing to internal components.

- **Dimensions and Weight Distribution:**
  We have revised the chassis dimensions to 750 x 500 x 190 mm to enhance the robot's maneuverability. The new design results in a reduced turning radius, which is critical for navigating and avoiding obstacles in tight spaces. Additionally, we repositioned the wheels closer to the center of the chassis, thereby improving the center of gravity. This adjustment significantly reduces the risk of toppling and enhances the overall stability of the robot.

- **Custom Mounts and Components:** Another major improvement was the in-house development of custom mounts and parts. Using FDM 3D printing, we designed and fabricated lightweight, accurate mounts for critical electronic components such as the LiDAR sensor and RealSense camera. We also used 3D-printed clamps to securely join the acrylic panels, ensuring precise alignment and a clean finish.
  Apart from 3D printing, we also utilized advanced machining techniques to produce custom wheel shafts and motor mounts tailored specifically to our bot's requirement. These custom components helped us optimize performance, reduce assembly issues, and maintain a compact and efficient design.

4.3 Drive Train

Qubi operates on a differential drive train system, which provides a simple yet effective method for maneuvering the robot with precision. The drive system is powered by two brushed DC motors that drive a pair of 10-inch wheels. These wheels are connected to the motors using custom-fabricated shafts and mounted securely to the chassis using in-house designed motor mounts.
All structural components of the drivetrain, including the motor mounts and shafts, were manufactured in-house. This allowed us to ensure high precision, optimal fit, and better integration with the overall chassis design.
One of the key advantages of this system is its modularity. The current 2-wheel drive setup can be easily upgraded to a 4-wheel drive configuration by simply installing two additional motor mounts in pre-designated mounting slots on the chassis. With minimal mechanical changes and a small update in the control code, the system can seamlessly switch to a 4-wheel drive setup, offering flexibility for future performance upgrades or changes in terrain requirements.

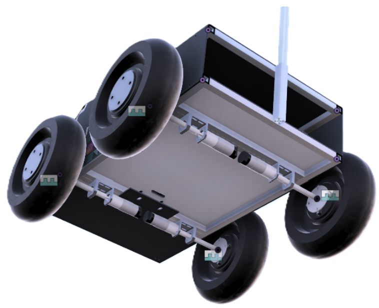


Figure 3: 4 Wheel Drive

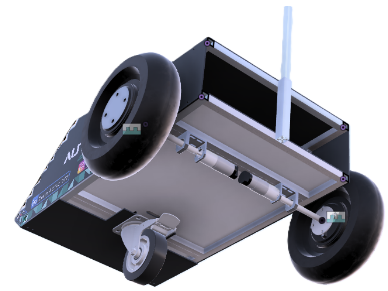


Figure 4: 2 Wheel Drive

4.4 Weather Proofing

We designed Qubi with weather resistance in mind to ensure consistent performance across different operating conditions. The chassis is enclosed with lightweight, durable acrylic panels that not only protect internal components from dust, moisture and UV rays but also give the robot a refined finish. To further safeguard the electronics, we sealed all the edges and the joints with rubber seal, effectively blocking out dust and preventing light water ingress. With these measures in place, Qubi achieves an IP62 rating, making it well-suited for semi-outdoor and industrial environments where exposure to dust and moisture is a concern.

5 Electrical and Electronics Design

5.1 Overview

The electronics of Qubi are designed on a dual-PCB structure to regulate and distribute power efficiently, isolating high-current motor lines from low-power peripherals. An STM32F103C8T6 microcontroller is employed for real-time motor control and encoder feedback, while high-level processing is managed by the Jetson Orin Nano. Brushed DC motors, MDDS30 motor drivers, and WCS1700 Hall-effect sensors are some of the principal components. Safety is implemented using inline fuses, an MCB, and redundant E-Stop systems (mechanical and wireless). The design focuses on modularity, thermal management, and real-time fault detection.

5.2 Significant Power and Electronics Components

Qubi is powered by a 6S 10Ah 22V Li-ion battery pack. Two custom PCBs handle power distribution:

- High Current PCB: A dedicated buck converter that steps down 22V to 18V, used for powering the brushed DC motor drivers.
- Low Current PCB: Supplies regulated 5V and 12V lines for peripherals such as the E-Stop board, control board, fans, and signal lights.

Main electronic components include:

- Motors: 2 × 18V brushed DC motors with inbuilt quadrature encoders.
- Controller Board: STM32F1 ("Bluepill") microcontroller, executing PID-based motor control based on encoder feedback.
- Motor Driver: Controlled using PWM and direction signals from the controller board.
- Current Sensor: WCS1700 Hall-effect current sensor for real-time motor current monitoring.

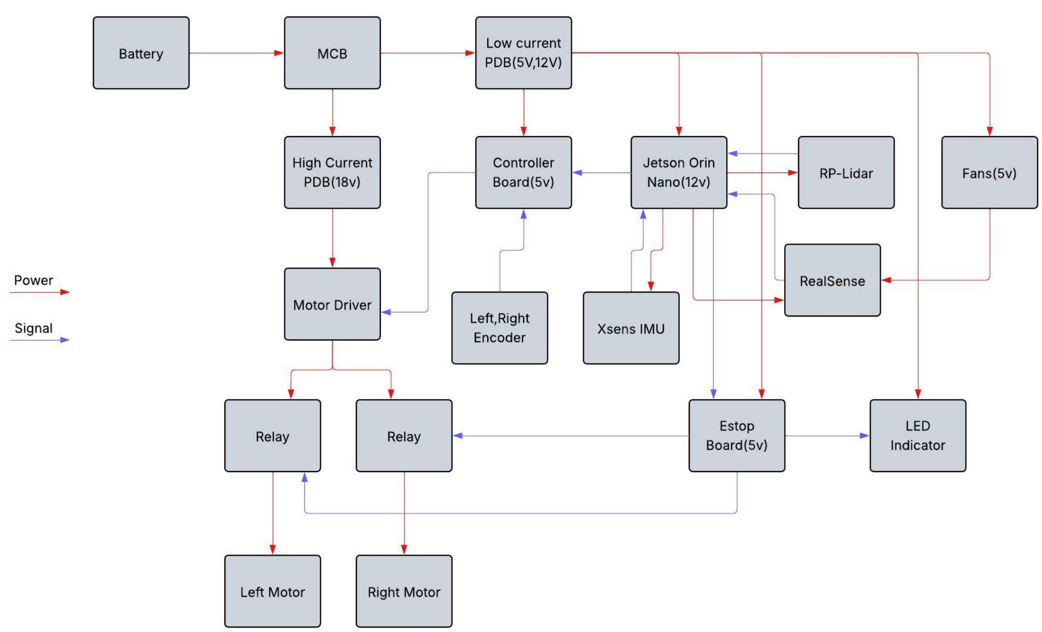
Thermal Management:

- To prevent thermal throttling of the Intel RealSense D435i, a dedicated cooling fan is mounted alongside the sensor. This addition was made after identifying overheating issues during last year's IGVC. Powered by the low current PCB, the fan ensures consistent thermal performance and reliable operation of the depth camera throughout the runtime.

5.3 Power Distribution System Specifications

The power distribution system is designed to isolate high-current and low-current paths for improved efficiency and safety. Key specifications are:

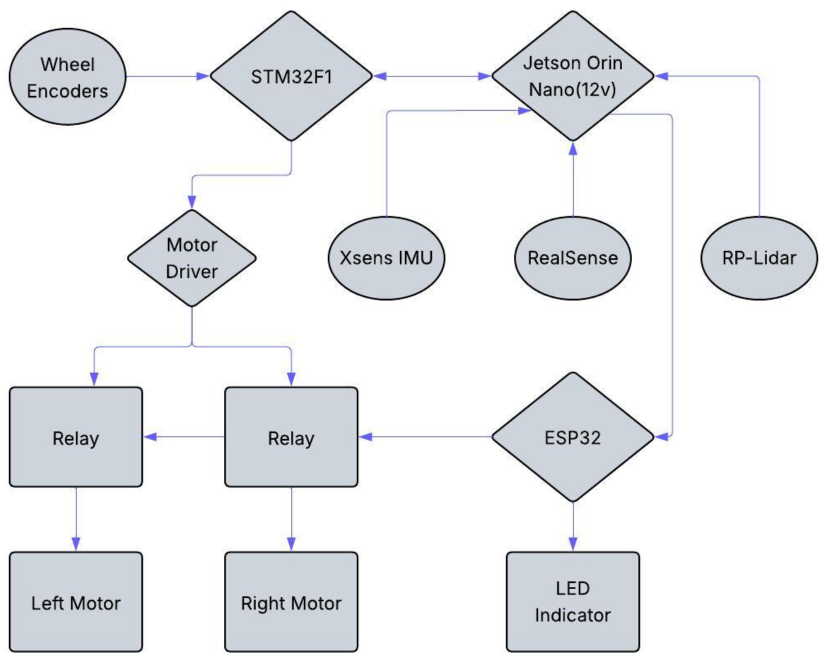
- Battery: 6S Li-ion, 22V, 10Ah
- Runtime: 45 minutes under normal operating conditions(0.5m/s)
- Recharge Time: Approximately 2hrs.
- Converters:
  - 22V → 18V (High current for motors)
  - 22V → 5V & 12V (Low current for peripherals)
- Safety Features:
  - Inline fuses on both PCBs
  - Real-time current monitoring with the WCS1700 sensor
  - The Emergency Stop system is designed to instantly disconnect power to the motor driver through a dual approach: a hardwired switch and a wireless LoRa-based controller.



5.4 Electronics Suite

Qubi integrates a wide variety of sensors and processing units for navigation, localization, and perception:

- Controller Board:
  - STM32F103C8T6 (Bluepill)
  - USB interfacing with Jetson Orin Nano for receiving velocity/Direction commands
  - Runs a PID controller based on encoder feedback
- Sensors:
  - Intel RealSense D435i – For lanes/Pothole detection.
  - Xsens MTi-670G – IMU with GNSS for localization and GPS.
  - RPLIDAR S2 – 360° 2D LiDAR for obstacle detection.
- Motor Driver:
  - Controlled via PWM signals from the STM32 controller board based on processed encoder feedback

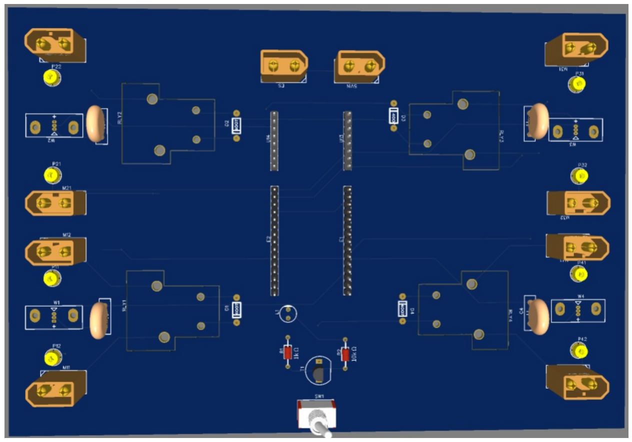


5.5 E-Stop Systems

The robot features a dual E-Stop system to ensure safety during both manual testing and autonomous operation:

- **Mechanical E-Stop:** Hardwired, relay-based cutoff switch physically mounted on the vehicle to immediately disable the motor power.
- **Wireless E-Stop:**Wireless E-Stop: Utilizes a LoRa-based wireless link between the handheld switch and onboard E-Stop board. Activation triggers a relay that cuts motor power.

- **Fail-safes:** The relay-based architecture ensures that any signal loss or LoRa failure defaults to a safe motor cutoff state.
- To enhance system safety, the **WCS1700 Hall-effect sensor monitors motor current** in real-time and initiates a shutdown of motor power in the event of abnormal current draw.



6 Software Design

6.1 Overview

In our experience testing various software frameworks for autonomous ground vehicles, we found that most of the plug-and-play solutions forced us to spend hours pouring over documentation without understanding the implementation, only to have something break at the smallest update. This time, our focus has been on keeping the solution as simple as possible, writing as much of the software and implementing as much of the algorithms as possible ourselves.

This approach has allowed us to gain a much better and deeper understanding of the various algorithms and nodes and how they fit together, enabling us to track down any bugs or issues and fix them quickly, greatly improving our efficiency and reducing our time spent identifying which change broke the system.

All the software for the bot runs on a Jetson Orin Nano controlled over SSH. When used to its full potential, it is a powerful and efficient solution to all our computing needs. This removes the need for multiple laptops that have to be charged and maintained with a clean working environment. The bot has multiple nodes written in Python/C++, implementing various parts of the Autonomy stack, giving us a modular design in which any one part could be swapped out for another algorithm without affecting any other parts. ROS2 humble has been essentially employed as a communication framework between all the different nodes and to take data from sensors and command the motors.

Figure 5: ROS Nodes

6.2 Perception

The perception stack includes a 2D LIDAR (RPLidar-S2), used for detecting obstacles, and a depth camera (Realsense D435i) primarily used for detecting lanes and potholes, determining the traversable parts of the ground plane.
The LIDAR, with a range of 30 meters (98 feet) and high sunlight tolerance, provides us with an extremely accurate 2D slice of its plane at 10 Hz. It is positioned low enough that it is able to capture all the obstacles present in the IGVC course. The data is acquired through the RPlidar ROS SDK that is provided and published to the */scan* topic.
The realsense depth camera is mounted on a pole, ensuring that we are able to make the most of its FOV. The frames received are converted to HSV and thresholding is performed to extract the areas that are white in color. The points in this binary mask are deprojected to arrive at a 3D point in the camera frame before running them through the inverse static transform to get the coordinates in the *base_link* frame. The point classified as being part of a pothole or lane only if the z-coordinate is low enough.

6.3 Localization and Mapping

Our custom localization solution uses data from three sources:
1. Wheel velocities from encoders built into the motors
2. Acceleration data from Xsens IMU
3. GPS data from the Xsens IMU
Data from all sources are fused together using a custom implementation of an Extended Kalman Filter (EKF). One node takes care of the continuous local position estimate using the acceleration and velocity data and another node listens for the discrete GPS data and corrects the global pose estimate each time.
The transforms are set up as follows:

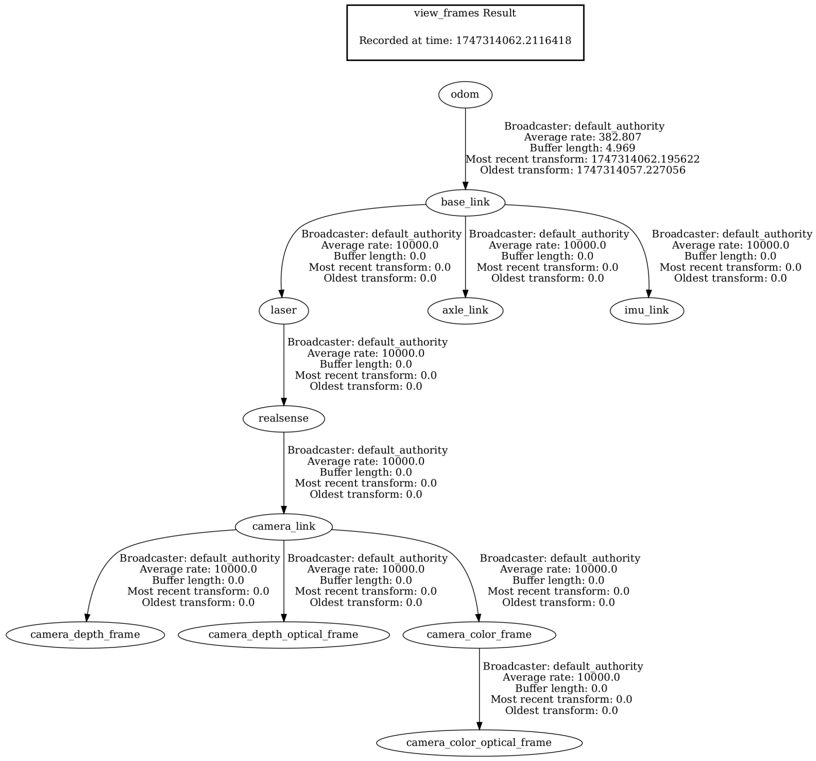


Figure 6: TF Frames

We have a custom mapper node that utilizes OctoMap in the backend. The mapper node listens for processed data from both the realsense and the LIDAR, constructing a 2D map. A kernel based dilation operation is then applied in order to add some inflation to the objects, giving us a simple way to ensure that the paths planned have sufficient separation from the lanes and obstacles. A gaussian kernel is then run over the map to indicate that areas close to the (inflated) obstacles too are better kept away from. This map is then converted to an OccupancyGrid message and published to the */grid_msg* topic.
We noticed that populating the OccupancyGrid was a bottleneck and parallelized the process using CUDA on the Jetson, enabling us to publish the messages at the rate of scan of the lidar itself.

6.4 Path Planning

Our path planner of choice is **D* lite**. It offers us a perfect balance of speed and optimality in path formed. It is perfect for the frequent re-planning necessitated by an environment that is still being discovered.
At each section, the next GPS waypoint is given as the goal and D* lite is able to plan a path towards it, avoiding any obstacles marked in the OccupancyGrid and replanning whenever a new obstacle is found. This path is then published to the */path* topic as an array of poses. **Model Predicitive Path Integral (MPPI)** is used as a local planner avoid obstacles that appear faster than the global planner can re-plan.

6.5 Controller Design

Once the path is published, our custom implementation of the pure-pursuit algorithm is employed to follow it. At every iteration, the algorithm looks at where the bot currently is and draws out a circle with a dynamic radius to find where it intersects the path. That point is then taken as the temporary target and the linear and angular velocities of the bot are set accordingly.
The look-ahead distance has been tuned to ensure that even tighter turns can be followed with a high level of accuracy. It is varied according to what speed the bot is moving at and what kind of region of the path it is currently at.

6.6 Simulation and Testing

We set up a **Gazebo** world with a bare-bones version of the IGVC track on it. A basic urdf file of our bot with functional components was generated and included in the world. The sensor plugins provided by Gazebo were used, seamlessly integrating with ROS in order to simulate, test and improve our algorithms and implementations.

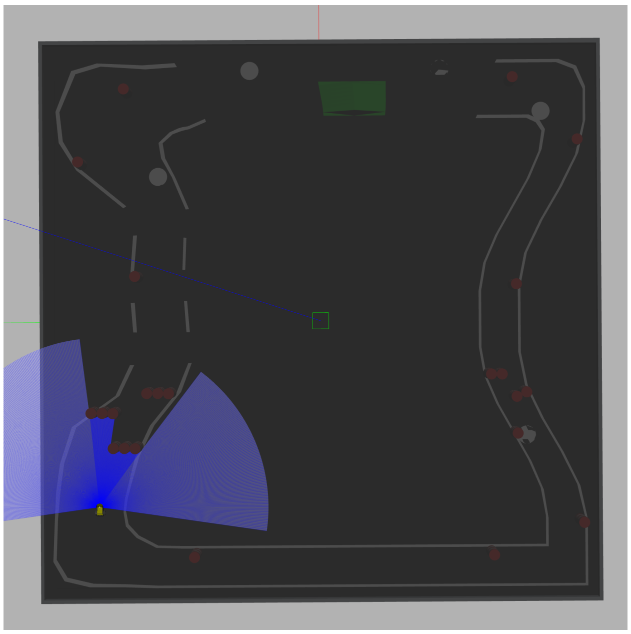


Figure 7: Simulation Top View

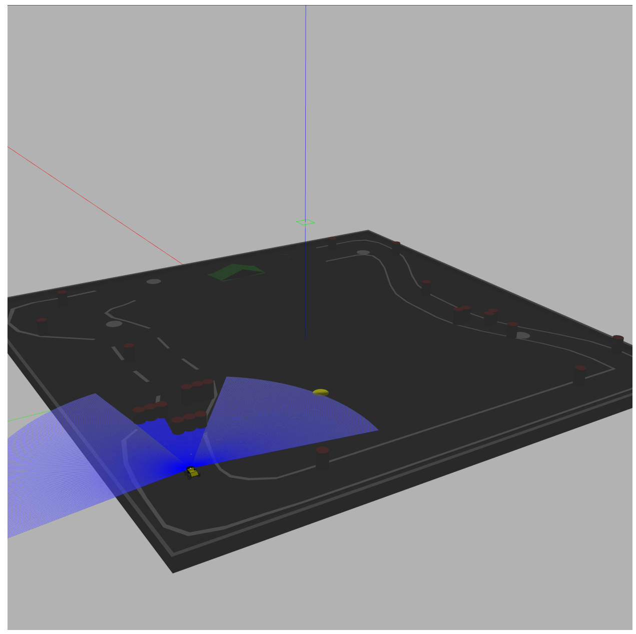


Figure 8: Simulation Side View

7 Cyber Security Analysis using RMF

7.1 Description of the NIST Risk Management Framework

Describe the NIST Risk Management Framework The NIST Risk Management Framework (NIST RMF) is a 7 step process that helps any organization or technology to manage and overcome cyber threats. The 7 steps can be explained as:

- **Prepare:** Establish context and assign roles to manage security and privacy risks. This step ensures that a risk management strategy is established and sets up governance.
- **Categorize:** Define the system and assess the potential impact of loss of confidentiality, integrity, and availability. This helps prioritize security efforts.
- **Select:** Choose appropriate security controls based on the risk level and system categorization using resources like NIST SP 800-53.
- **Implement:** Apply the selected controls in the system and document how each one is configured and used.
- **Assess:** Evaluate whether the controls are correctly implemented and functioning as intended to reduce risk.
- **Authorize:** A senior official reviews the assessment results and determines if the system's risk level is acceptable for operation, thereby providing accountability.
- **Monitor:** Continuously track and respond to changes, vulnerabilities, or threats, and regularly assess the effectiveness of controls.

7.2 Modelling Threats and Analyzing Impacts

System	Description of Threat	Confidentiality	Integrity	Availability
Source Code	if source code has been tampered with and malicious code has been executed intead	moderate	high	high
ROS Commands	if rival teams sends messages on topics that our bot listens to, they can alter the expected behaviour of the robot	low	high	high
Wireless E-Stop	if the E-stop signal spectrum is jammed or if rivals send signals on the same spectrum, bot can stop in the middle of track	low	high	high
Version Control Credentials	if rivals have access to the team's github repo, they can tamper with code, etc.	high	high	low

7.3 Cyber Controls and their Implementation

Most of Qubi's computation and control is done by the on board Jetson Orin Nano, and all components on the bot are connected via physical wires. This reduces the risk of cyber threats to a great extent. When using SSH to access the bot's Jetson, several NIST RMF control families apply to ensure secure and authorized access. From the Access Control (AC) family, controls like **AC-2** and **AC-17** ensure only authorized users can access the system remotely, while **AC-6** enforces the principle of least privilege. The Identification and Authentication (IA) controls, such as **IA-2** and **IA-5**, emphasize the use of strong authentication like SSH key pairs and proper key management. The System and Communications Protection (SC) family addresses secure data transmission through controls like **SC-12**, **SC-13**, and **SC-28**, which mandate encryption for data in transit. Additionally, **SC-7** ensures boundary protection by restricting SSH access to trusted networks. Finally, the System and Information Integrity (SI) controls, especially **SI-4** and **SI-7**, support monitoring SSH login attempts and maintaining updated, secure software.

8 Analysis and Failure Modes

Our team's previous bot, Pratham, had multiple issues which caused complications during IGVC 2023. Here are some of the way in which we have mitigated these problems:

- Realsense depth camera used to heat up while running the bot, especially in direct sunlight for longer time periods. This issue was overcome by creating a custom mount for the realsense camera, with slits and inbuilt fans to provide necessary cooling.
- Difficulty in assembling and disassembling due to the usage of multiple L-clamps with aluminium extrusions. This also made our bot more prone to vibrations, further loosening the joints. This was remedied by choosing to weld mild steel pipes to form the base chassis of new bot Qubi. This leads to better structural stability and cuts out assembly time by a lot.
- The lidar used in Pratham, RP-Lidar A1, completely refused to work under heavy sunlight. So, this time we chose to upgrade to the newer RP-LIDAR-S2, which now gives us the ability to withstand even sunlight of 80 klux and improves our range from 12m to 30m with better angular resolution.
- The range of the wireless E-stop was quite low owing to the use of an NRF module. In this version, we switched to LoRa, giving us a significantly higher range and a more reliable connection.
- Last time Pratham had only one 6s22v 22Ah battery and due to airline regulations it couldn't be brought with us for the competition, which lead us to get a new battery with lower ratings. To overcome this issue, this year we got two set of custom made 6s 22v battery with 4.5Ah rating.

- Pratham made use of a laptop for running the entire software stack. This proved to be a hassle for testing and debugging. Also, the laptop ran out of charge often while testing. All these problems were solved by using Jetson Orin Nano. All computations were offloaded to the Jetson, which is powered by the custom power distribution system.

Point of Failure	**Cause**	**Failsafe Strategy**
Bolt loosening due to vibration	Due to Continuous mechanical vibrations during operation	Used spring washers and thread-locking adhesives to secure fasteners.
Loosening of bolts at L-clamp connections weakened the overall structural integrity.	Due to Continuous mechanical vibrations during operation	Replaced chassis material with mild steel (MS) and welded it into a single rigid, solid structure.
Wobbling of the Realsense pole	Lack of effective vibration isolation and inadequate pole mounting support.	Adding rubber dampeners to minimize vibrations and reduce pole wobbling.
Poor maneuverability	Unbalanced weight distribution and inefficient wheel layout	We redesigned the whole drivetrain- reduced the number of caster wheels from 2 to 1 and relocated it to the front and also repositioned the drive wheels towards the rear and closer to the center to achieve better weight distribution and enhanced maneuverability.

Table 1: Mechanical Failure Modes

Point of Failure	**Cause**	**Failsafe Strategy**
Sudden Motor speed spike/Motor Overcurrent	Loss of signal from controller board	Current Monitored using WCS1700, triggers the Estop.
Power supply failure	Voltage Regulators failure/overload	Backup-off the shelf Voltage converters
Realsense Heating	Prolonged exposure to direct sunlight	Dedicated cooling fans has been mounted to the sensor
Wireless Estop failure	Loss of signal between transmitter and receiver	Estop is triggered

Table 2: Electrical Failure Modes

Point of Failure	Cause	Failsafe Strategy
Odometry Drift	Small errors in velocity and acceleration accumulate to larger errors after integration leading to a drift in the bot's predicted and actual position	Extended Kalman Filter used to fuse IMU and wheel odometry data.
Noisy Camera Images	Sensor limitation of the depth camera	Point clusters that are too small are ignored
Incompatible Software Versions	Different libraries require different versions of the same software leading to dependency errors	Fixed by containerization and using virtual environments
Map Updation Bottleneck	The map updation was too slow in single threaded configuration	Sped up by using GPU capabilities of Jetson using CUDA

Table 3: Software Failure Modes

9 Initial Performance Assessments

Max Speed	3.98 meters/second (Hardware Limited) 1.5 meters/second (Software Limited)
Acceleration Limited by code	2 meters/second^2
Battery Life	24 hours on standby 1.5 hours with motors running
Ramp Inclination	10 degrees
Motor Current Limit	6A
Lidar Range	30 meters
Reaction Time	10Hz path updation

References

[NIST(2024)] National Institute of Standards and Technology (NIST), 2024, Risk Management Framework (RMF), https://csrc.nist.gov/projects/risk-management/about-rmf

[Open Robotics(2024)] Open Robotics, 2024, ROS 2 Humble Tutorials, https://docs.ros.org/en/humble/Tutorials.html

[OpenCV(2024)] OpenCV, 2024, OpenCV 4.x Tutorials, https://docs.opencv.org/4.x/d9/df8/tutorial_root.html