



Tribhuvan University
Institute of Engineering, Pulchowk Campus
Department of Mechanical and Aerospace Engineering
Pulchowk, Lalitpur

**DEVELOPMENT AND TESTING OF AUTONOMOUS MOBILE ROBOT
FOR MATERIAL HANDLING**

Project Supervisor: Assistant Professor Dr. Surya Prasad Adhikari

Presented By:

Nirmal Prasad Panta (076BME025)

Pawan Shrestha (076BME027)

Prince Panta (076BME029)

Saki Basnet (076BME036)



Outline

- Introduction
- Problem Statement
- Scope
- Objectives
- Literature Review
- Methodology
- Output
- Work Completed
- Limitations
- Problems Faced
- Budget Analysis
- Work Schedule (Gantt Chart)
- Conclusion and Future Enhancement
- References



Introduction

- Agile corporations leveraging robots and autonomous technologies will thrive.[1]
- Automation has ability to tackle problems like labor shortage, safety risks.
- Flexible manufacturing systems can bring tremendous economic advantages, where AMR becomes of at most importance.
- AMRs transport raw materials, semi-finished products and products from inventory, job stations and assembly line to respective place.



Main Objective

To develop an autonomous mobile robot for autonomous transportation using Simultaneous Localization and Mapping (SLAM) and Nav2 for mapping, localization, and navigation and computer vision (CV) for Material handling



- To create an AMR prototype using conventional control for navigation
- To employ SLAM algorithm for mapping and Nav2 to enhance the AMR's autonomous navigation, precise localization, and effective obstacle avoidance
- To integrate computer vision technology to augment the AMR's material handling capabilities, enabling it to accurately identify and handle payloads.

Literature review

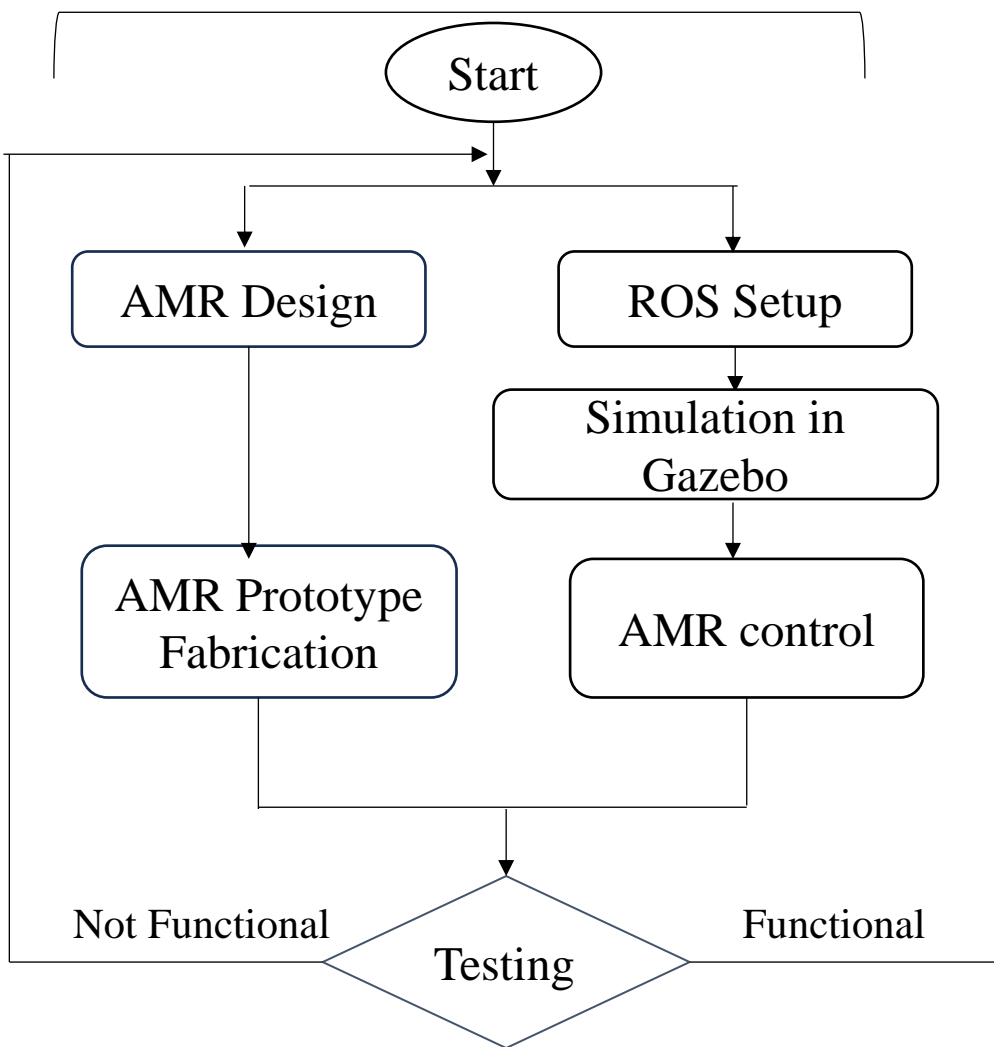


Evolution of AGV to AMR	G. Ullrich, The History of Automated Guided Vehicle Systems, 2015
	W. Grzechca, Manufacturing in Flow Shop and Assembly Line Structure, 2016
	N. Zghair et.al , A one decade survey of autonomous mobile robot systems, 2021
	L. Lynch et al., Integration of autonomous intelligent vehicles into manufacturing environments: Challenges, 2020
ROS2	S. Macenski et.al , Robot Operating System 2: Design, architecture, and uses in the wild, 2022 (ROS 2 Documentation in https://docs.ros.org/)
SLAM	S. Macenski and I. Jambrecic, SLAM Toolbox: SLAM for the dynamic world, 2021
Nav2	S. Macenski, et.al, The Marathon 2: A Navigation System, 2020 (Nav2 Documentation in https://navigation.ros.org/)
OpenCV	J. Wubben et.al, Accurate Landing of Unmanned Aerial Vehicles Using Ground Pattern Recognition, 2019
	G. Bradski, The OpenCV Library, 2000.

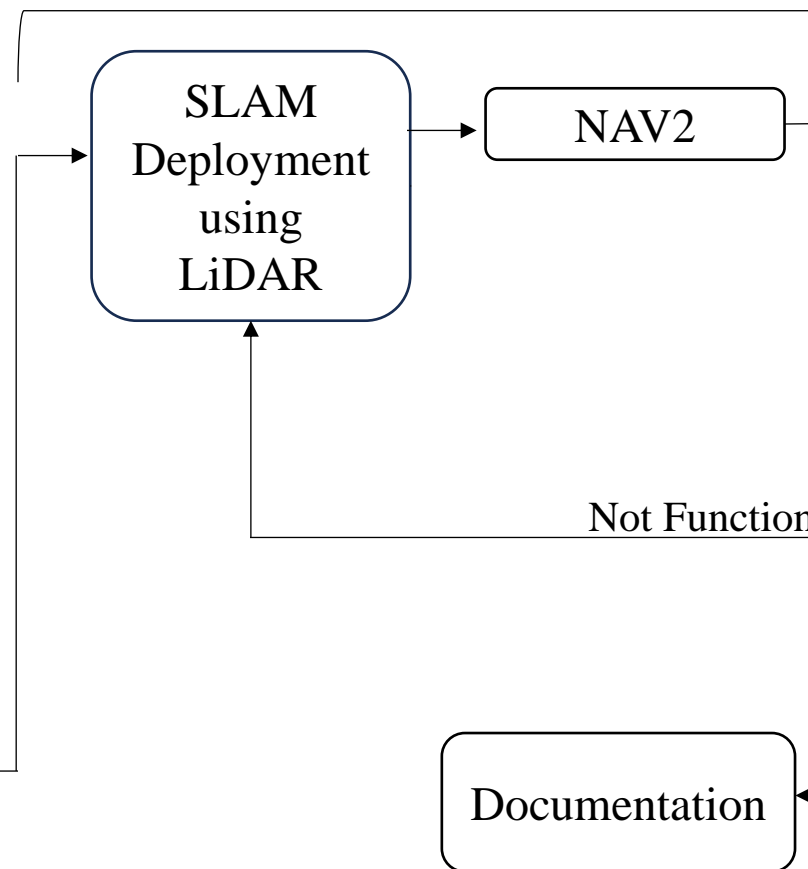


Methodology

Phase I: Transport



Phase II: Improvement on Transport



Phase III: Functionality to Transport

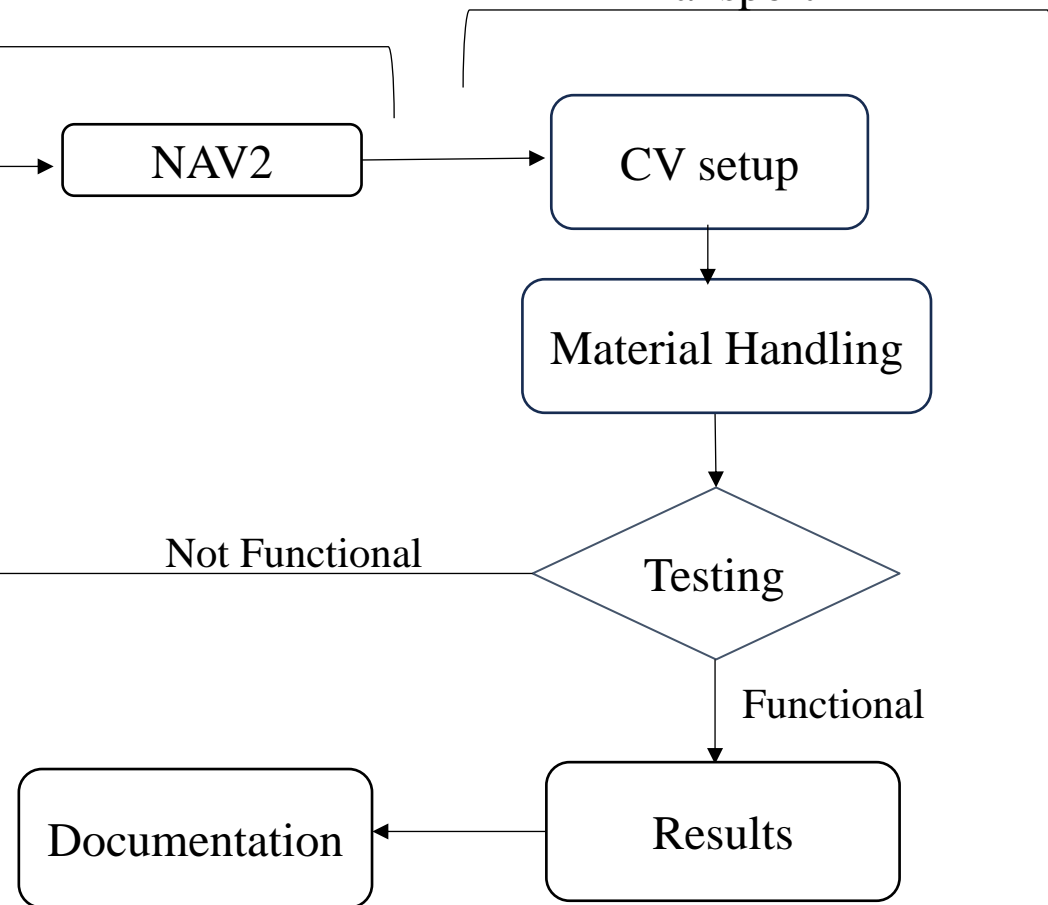


Figure 1: Methodology Flow Chart



AMR CAD Model

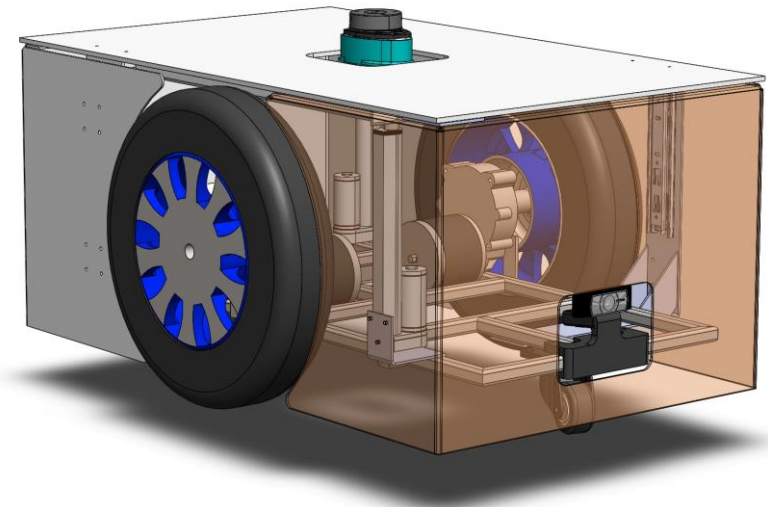
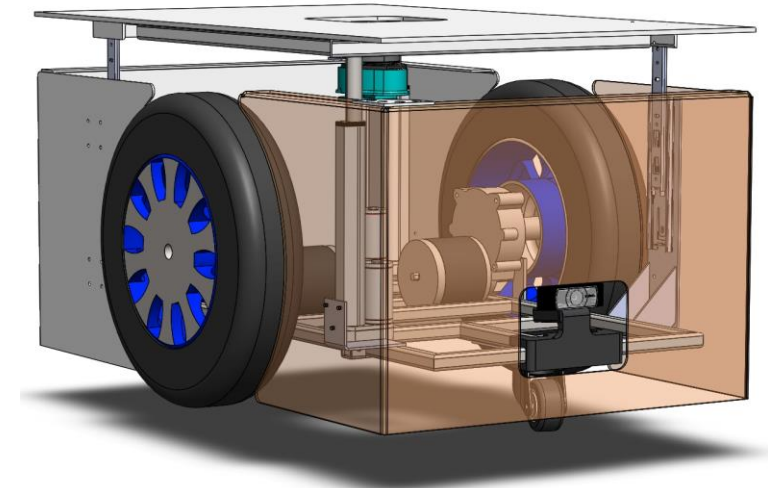
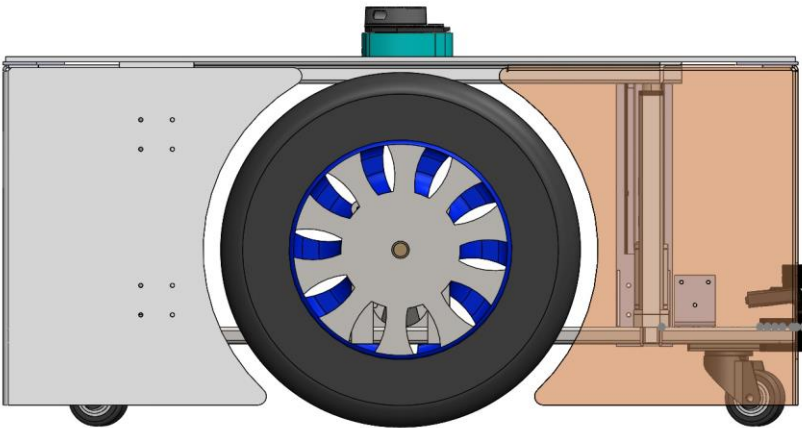
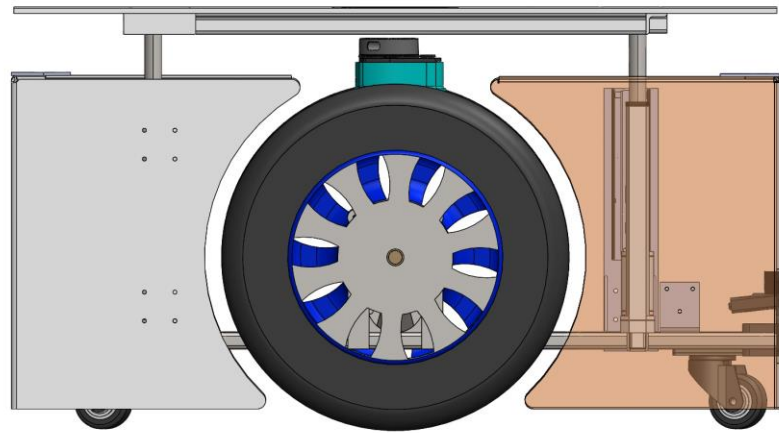
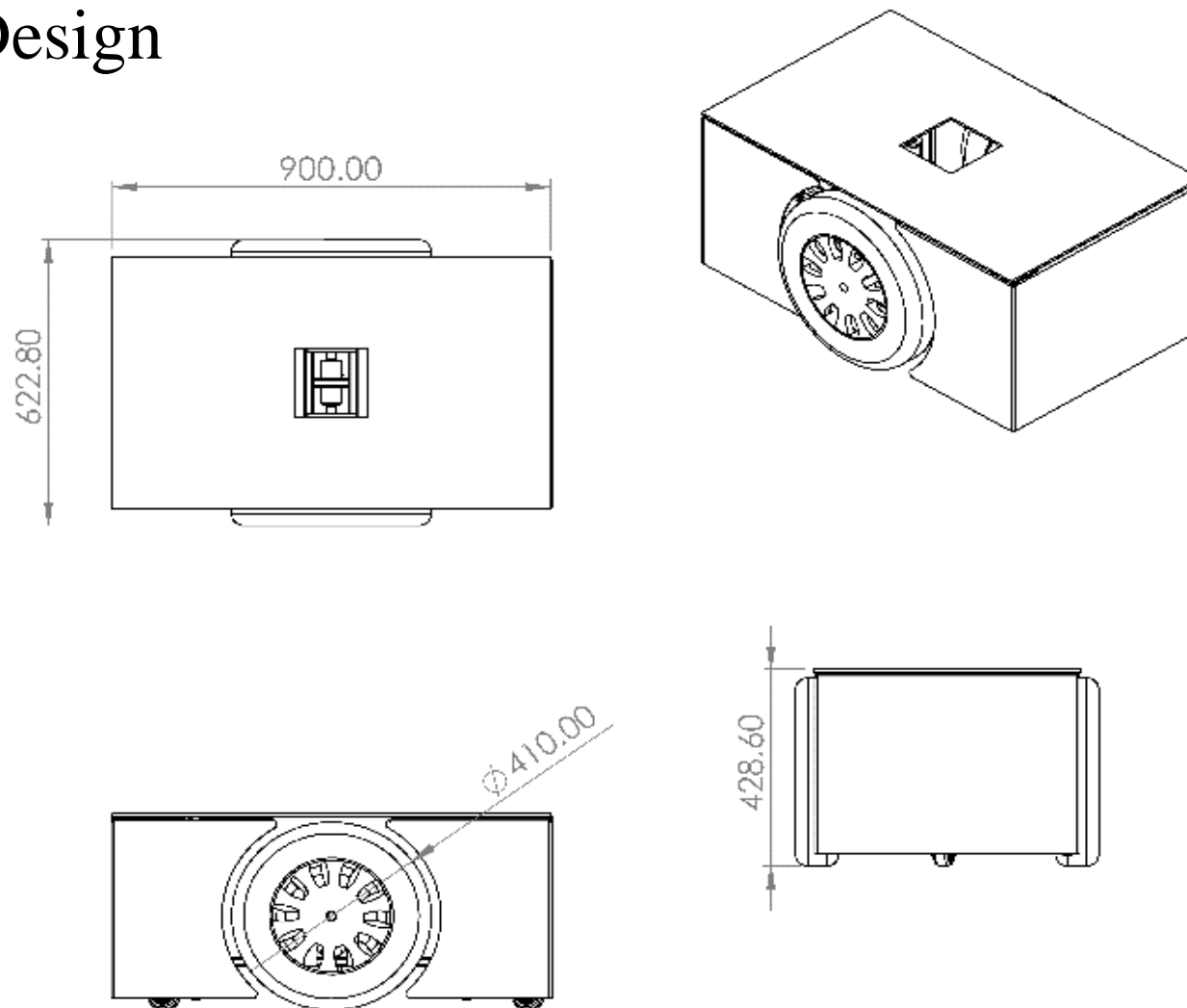


Figure 2: AMR prototype CAD model



AMR Design



Max payload capacity : 127.7kg
Motor torque : 125 kg.cm
Weight of robot : 30.8 kg

Figure 3 : AMR Dimensions



AMR Design and Analysis

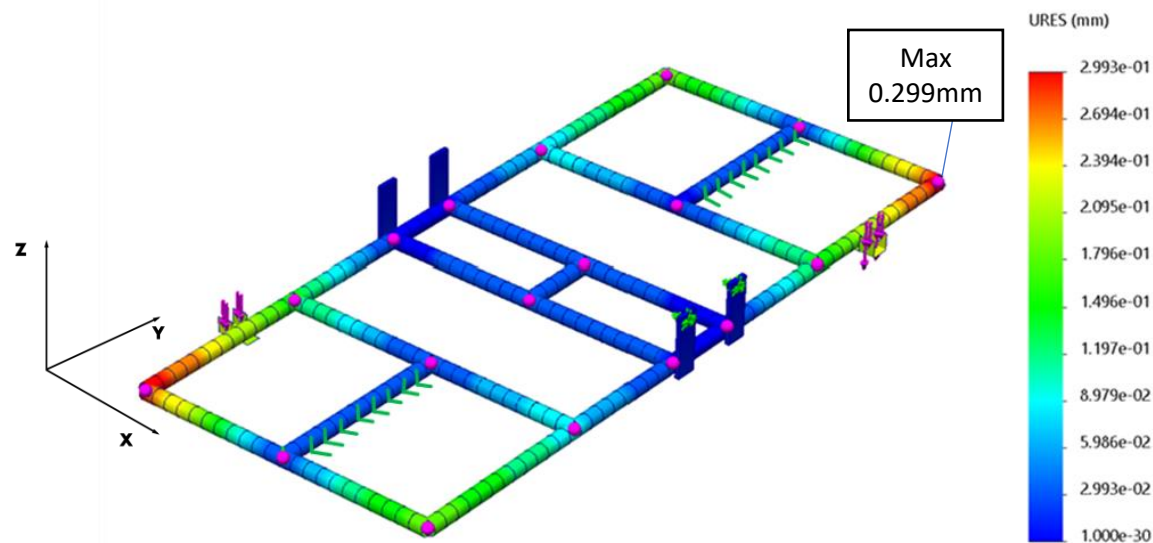
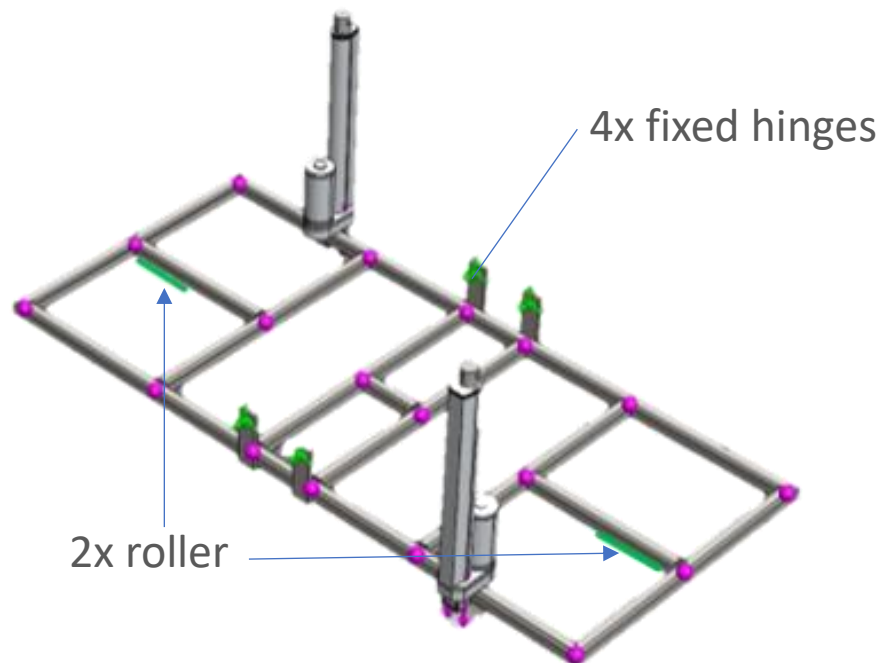
Table 1: Material Properties

Name:	Plain Carbon Steel
Model type:	Linear Elastic Isotropic
Yield strength:	2.20594e+08 N/m²
Tensile strength:	3.99826e+08 N/m²
Elastic modulus:	2.1e+11 N/m²
Poisson's ratio:	0.28
Mass density:	7,800 kg/m³
Shear modulus:	7.9e+10 N/m²

- Static structural simulation results show the frame deflecting under load, with maximum displacement of 0.2993 millimeters
- The maximum von Mises stress of 115.4 MPa
- The minimum Factor of Safety is 1.9



AMR Static Study



Load in -Z from each actuator: 50kg

Figure 4: Model reference and the displacement plot



AMR Chassis Fabrication



Figure 5: Chassis Fabrication



Simulation of AMR in ROS 2

- Package `project_amr` created and built with `colcon`
- Launch files for the simulation created
- Simplified robot model created with the help of `urdf` and `xacro`
- Gazebo reference and sensor also added in the `urdf` file for simulation in gazebo and RVIZ
- SLAM and Nav2 documentation utilized for mapping and navigation



Creation of Robot Model

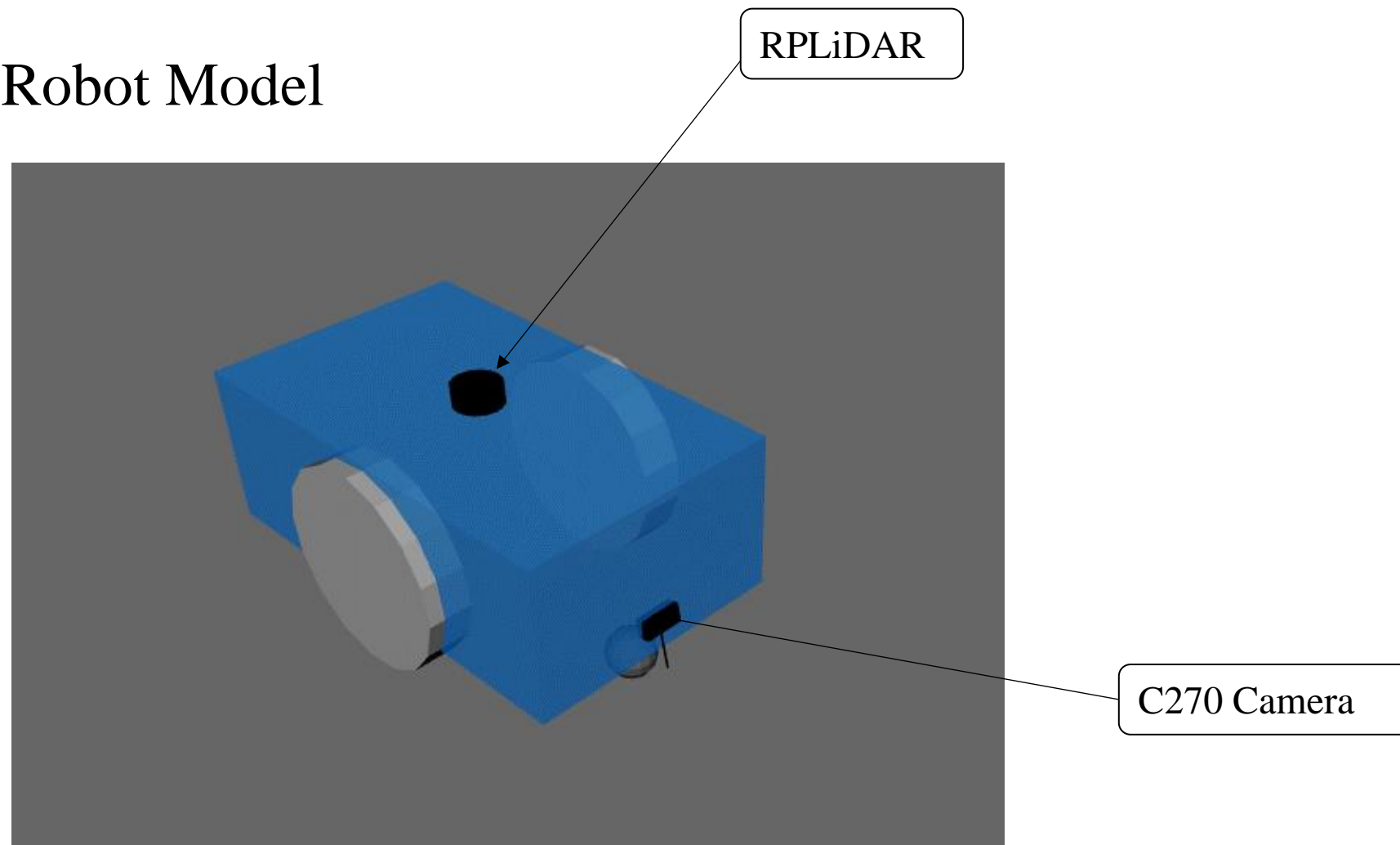


Figure 6 : Creation of simplified model for LIDAR and camera



Simulation of AMR in ROS 2

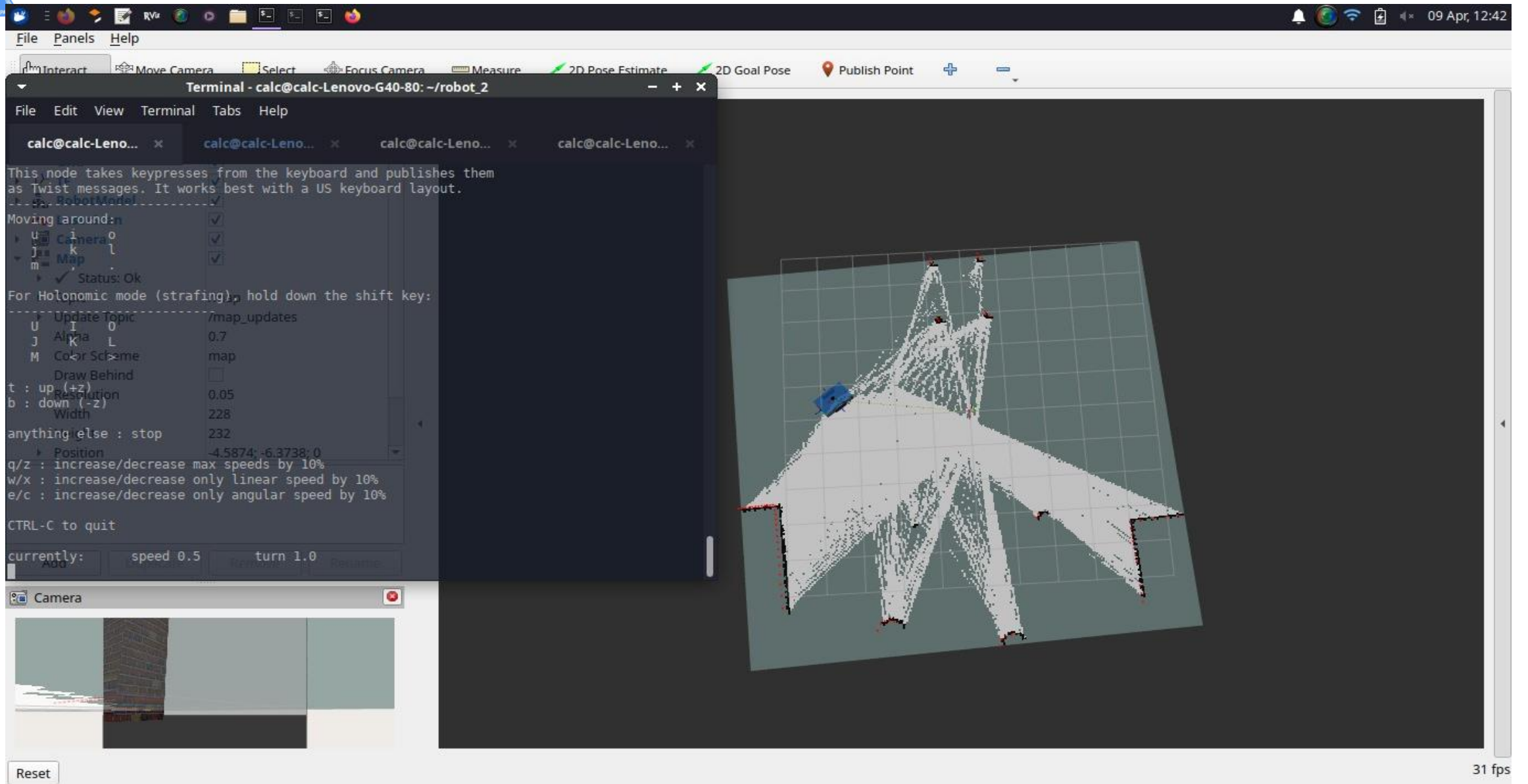


Figure 7: Creating map of virtual workspace using SLAM



Simulation of AMR in ROS 2

SLAM

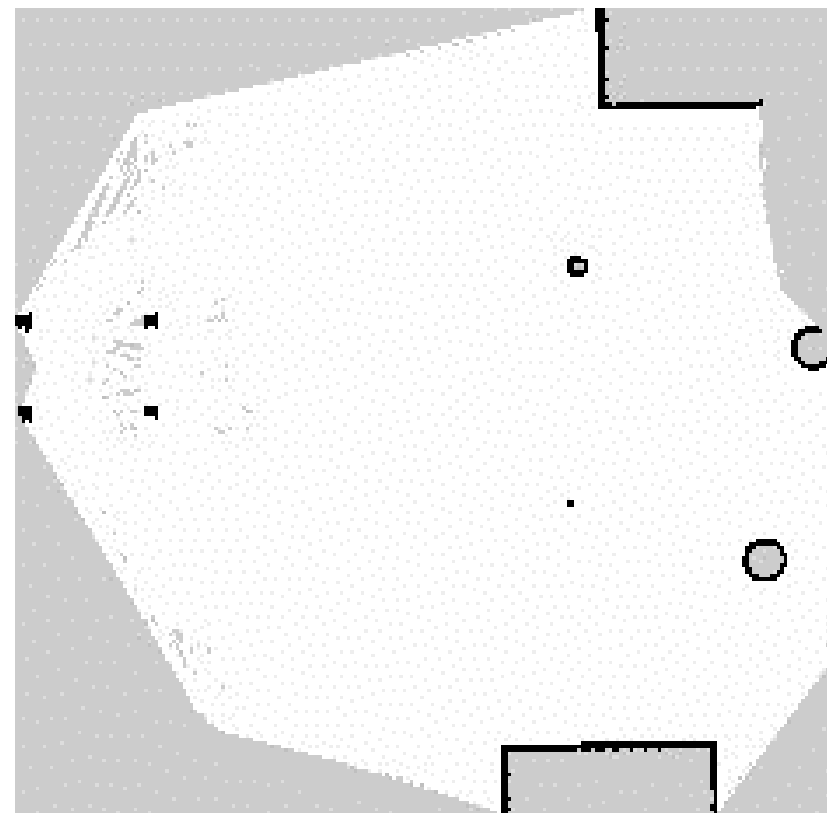
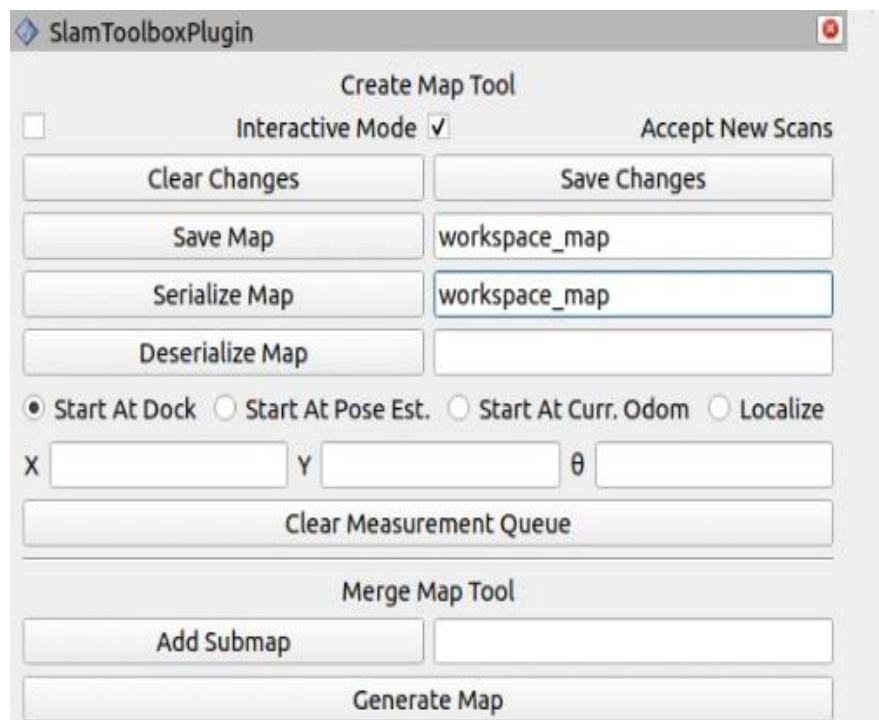


Figure 8 : Use of SlamToolboxPlugin to create workspace_map



Simulation of AMR in ROS 2

Nav 2

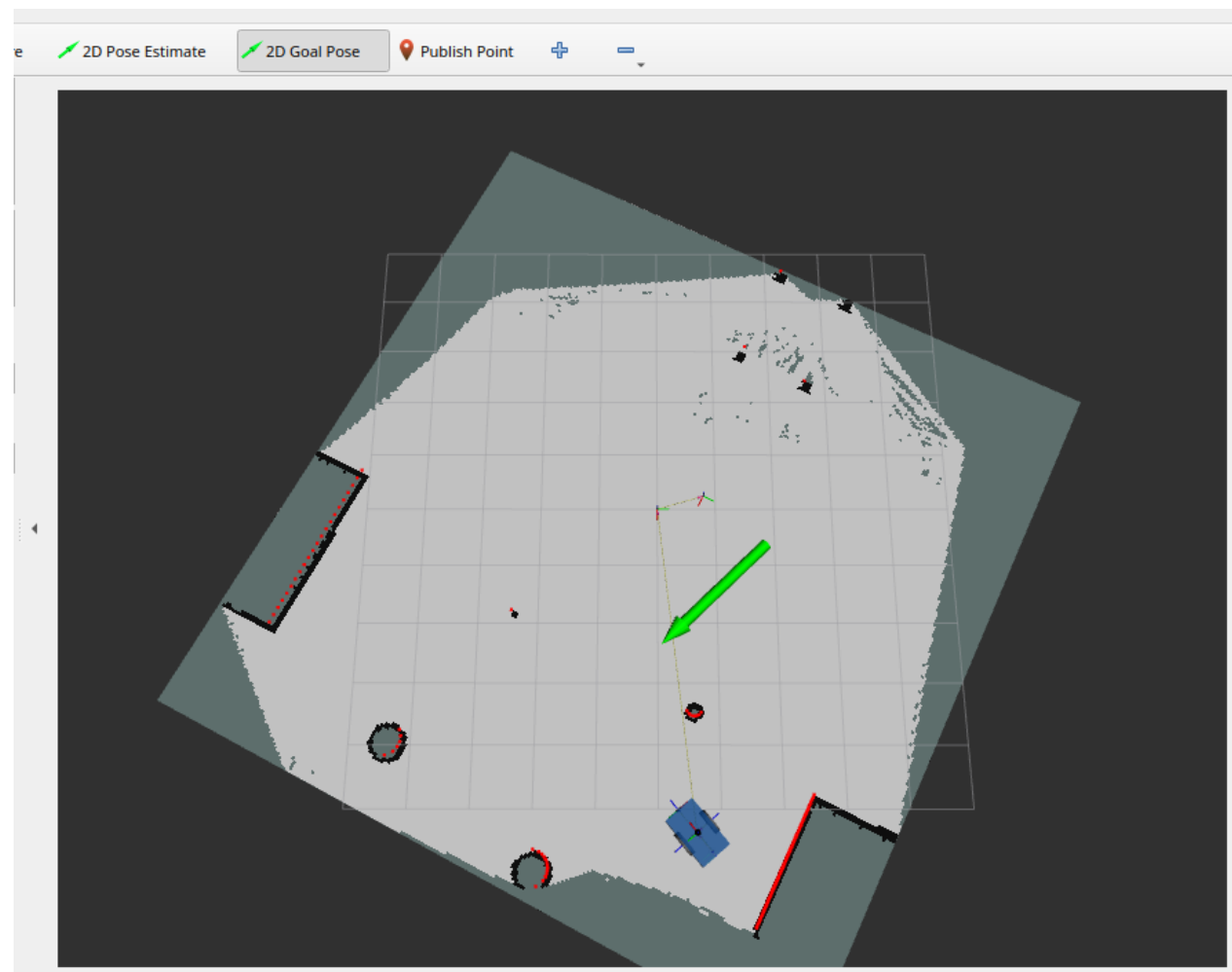


Figure 9 : Giving goal pose using Nav2



AMR Remote Control

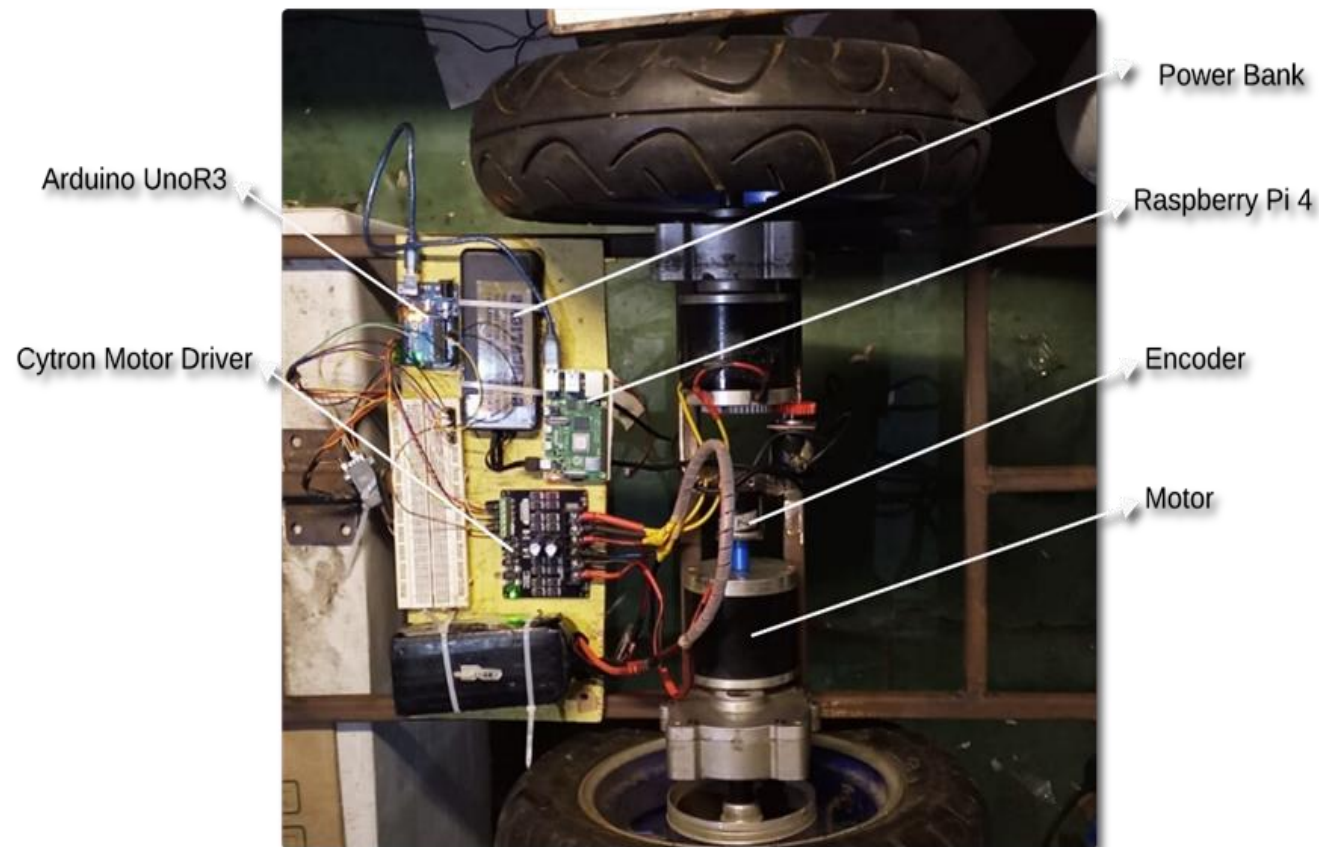


Figure 10 : Actual Control Circuit of AMR

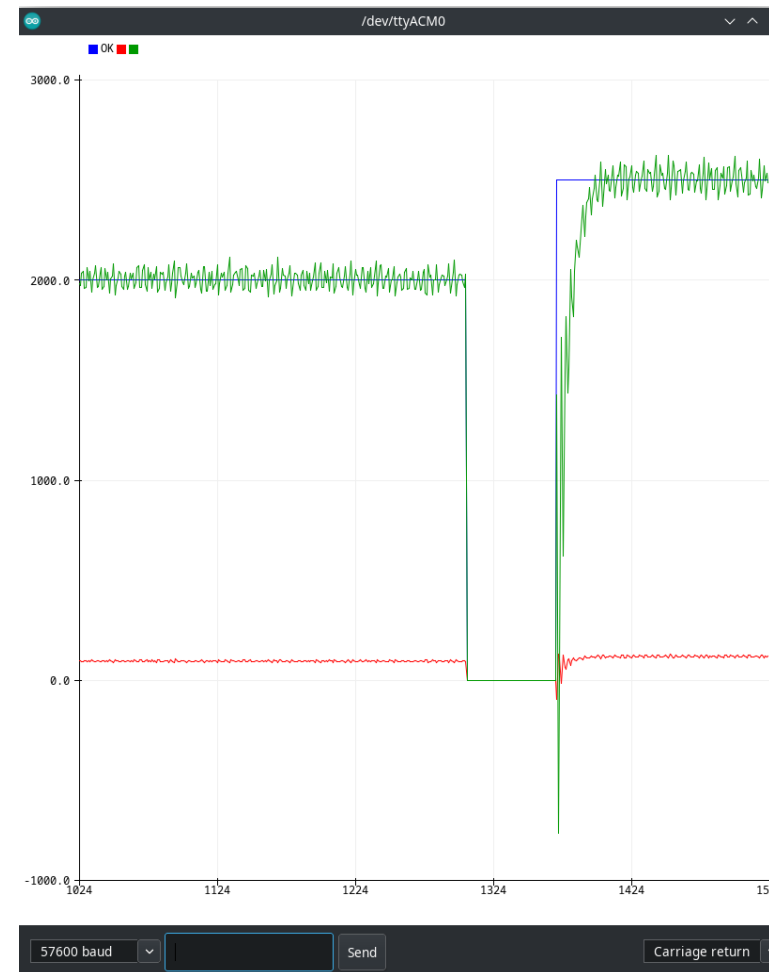
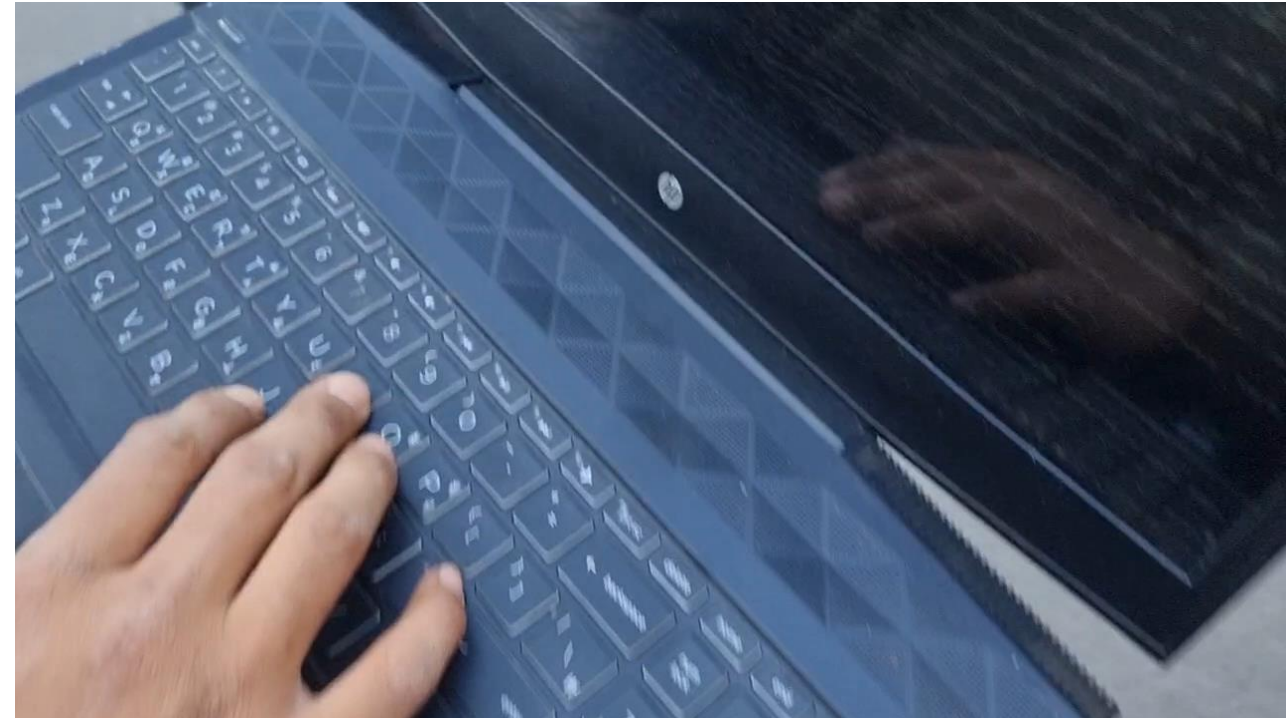


Figure 11 : Tuned PID response;
 $K_p=0.75$, $K_i=0.0001$, $K_d=4$ and $K_o=50$



Remote Controlled AMR Locomotion

- Two motors/actuators velocity controlled using **PWM** from **Cytron Motor Driver**
- Feedback from **Encoder** to enable **closed loop PID** control
- **Arduino** computes PID and uses **Encoder Counts per loop** as velocity variable at 30Hz.
- **Raspberry Pi** runs **ROS2 Control** that computes the required wheel velocity and corresponding Encoder Counts per loop and passes to Arduino using serial communication
- Server or another **PC** (for now) passes command to Raspberry PI about the task of the robot via **ssh**.
- **Command velocity** given from PC using either teleop_twist_ **keyboard** or the **joystick** (joy and teleop_twist_joy node)





ROS2 Control

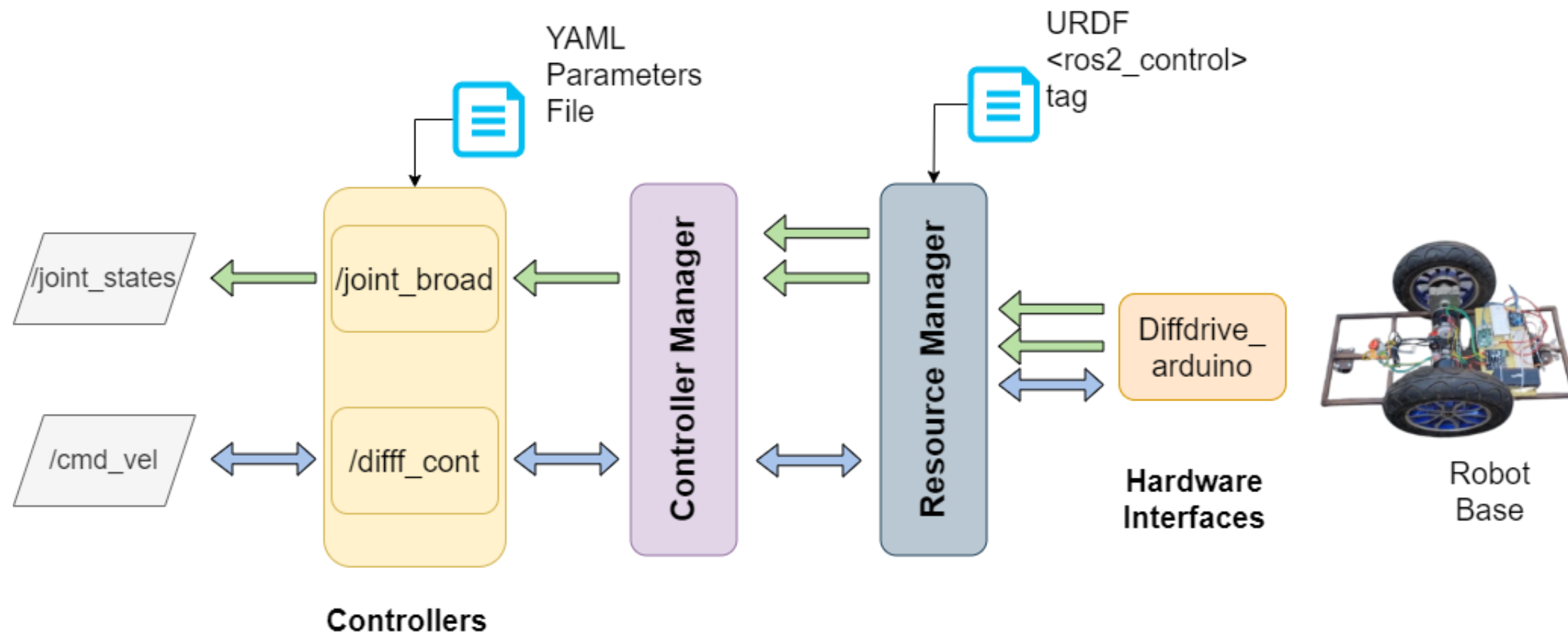


Figure 12: ros2 control architecture



ROS2 Control

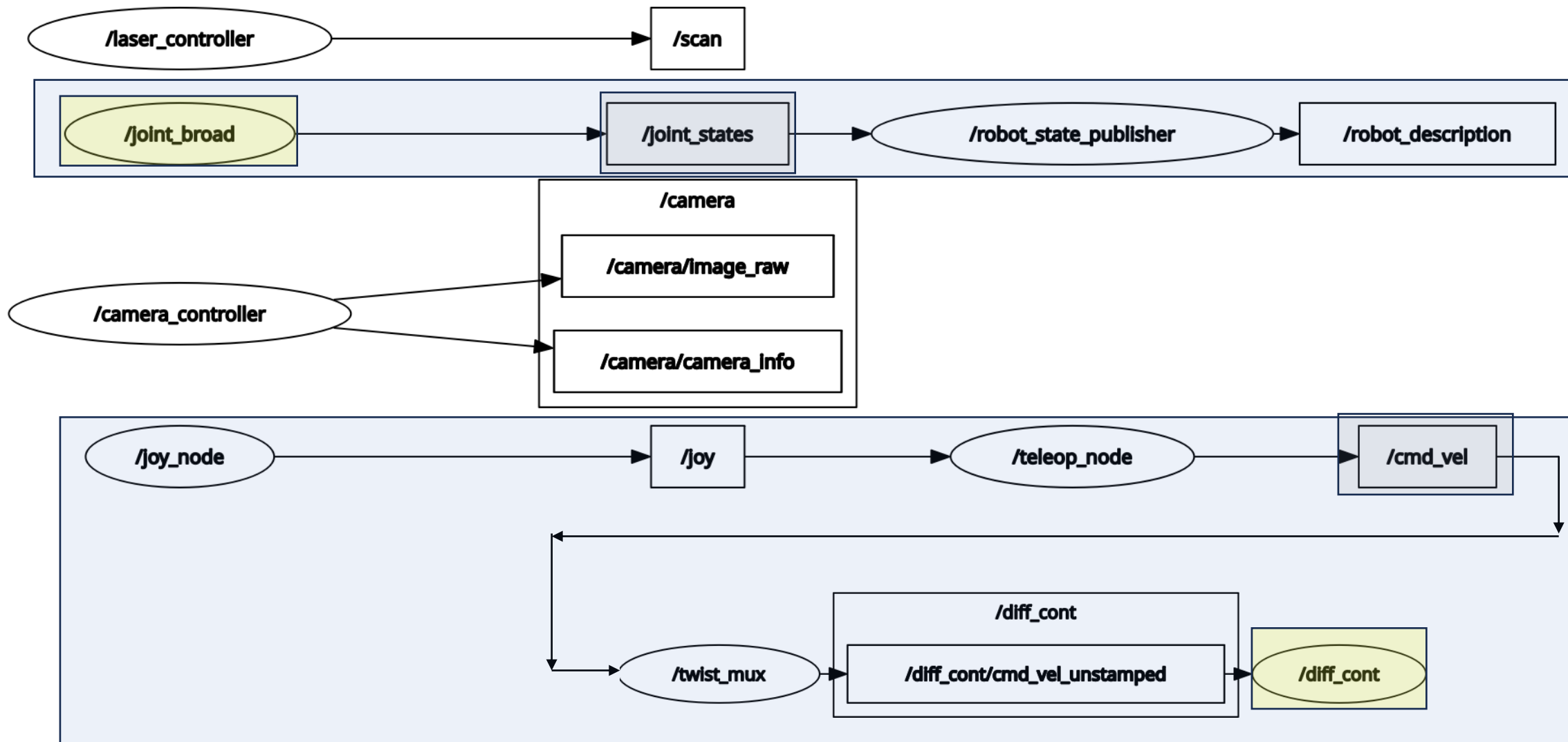


Figure 13: 8 Rqt_graph for ros2_control in simulation



Testing RPLIDAR

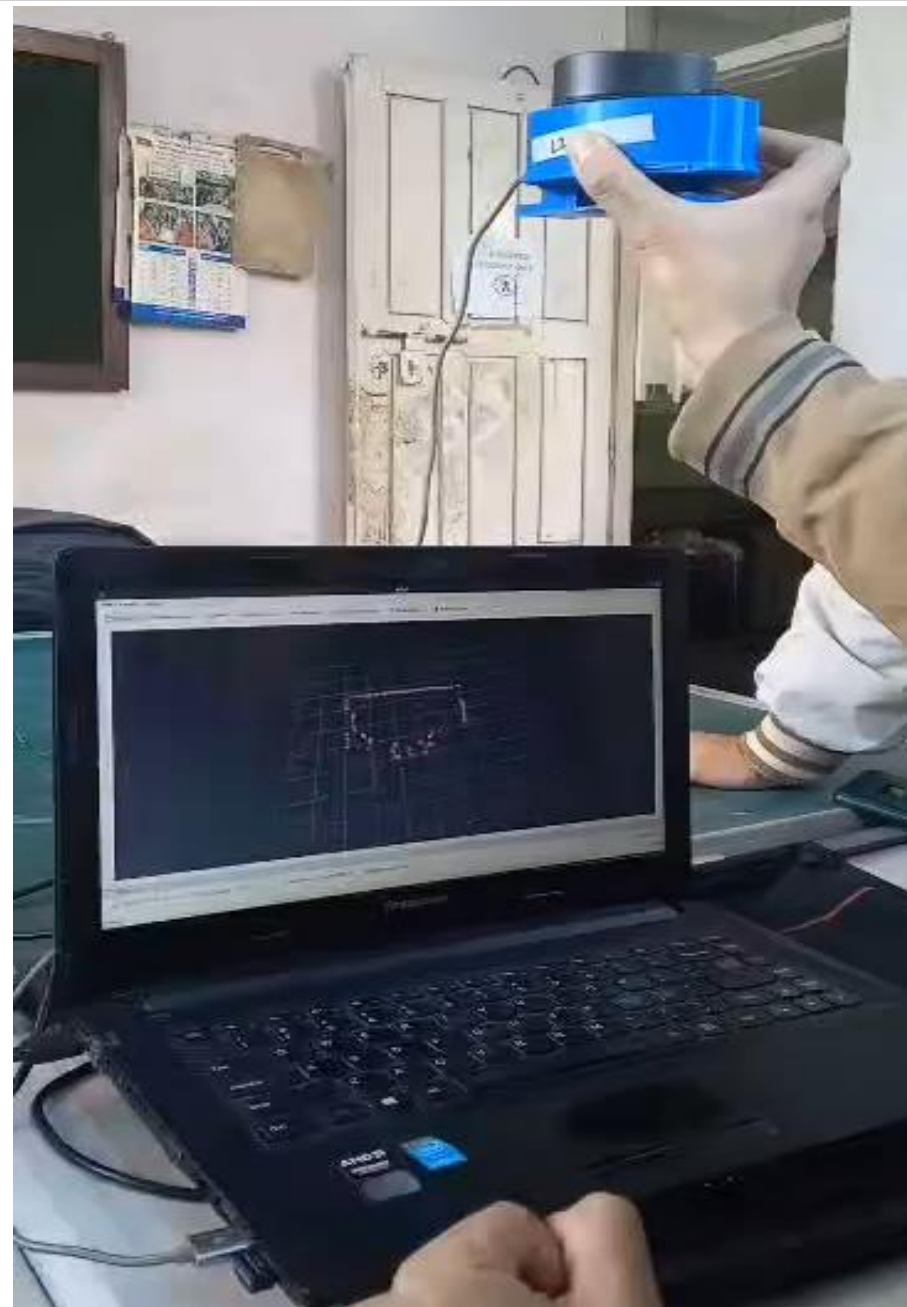
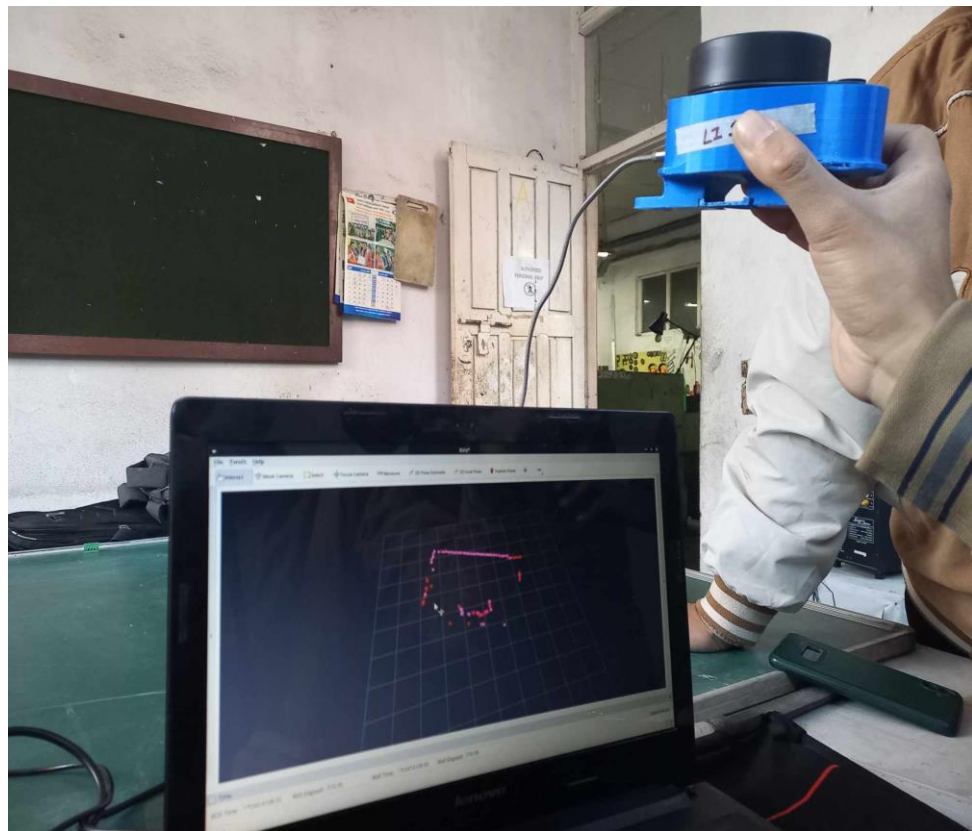


Figure 14 : Reading LIDAR's data using RVIZ



AMR Autonomous Navigation

- RPLiDAR generates the point cloud of surrounding and passes it to Raspberry Pi
- Server or PC computes SLAM, Navigation and publishes the command velocity for the required path.
- ROS2 Control's Diff Cont controller computes the desired AMR wheel velocity
- The hardware interface of ROS2 control communicates the required counts per PID loop via serial port.
- Closed loop control same as in Phase I

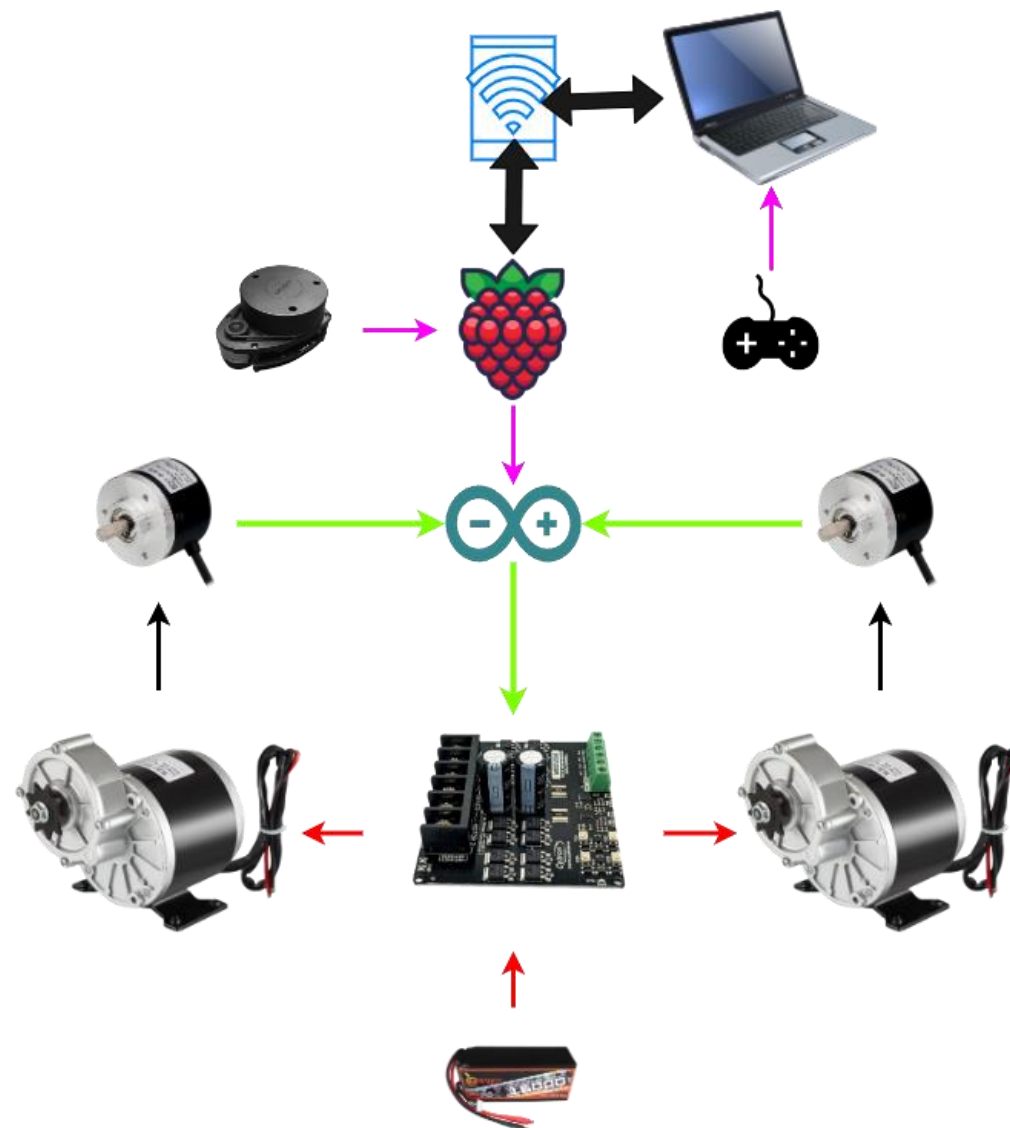


Figure 15 : Overall Navigation Control Pipeline



SLAM Deployment Using Lidar

RViz*

els Help

act Move Camera Select Focus Camera Measure 2D Pose Estimate 2D Goal Pose Publish Point + -

Enabled

Properties

Interval 0

1

tion Source Topic

tion Topic /robot_description

K

can

us: Ok

/map

Topic /map_updates

0.7

heme map

hind

on 0.05

188

281

-3.0526; -4.0085; 0

tion 0; 0; 0; 1

estamp

msg/OccupancyGrid topic to subscribe to.

Duplicate Remove Rename

1712563380.66 ROS Elapsed: 100.31 Wall Time: 1712563380.69 Wall Elapsed: 100.31

Experimental

31 fps

Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click/Mouse Wheel: Zoom. Shift: More options.



Map of workspace in Robotics Club

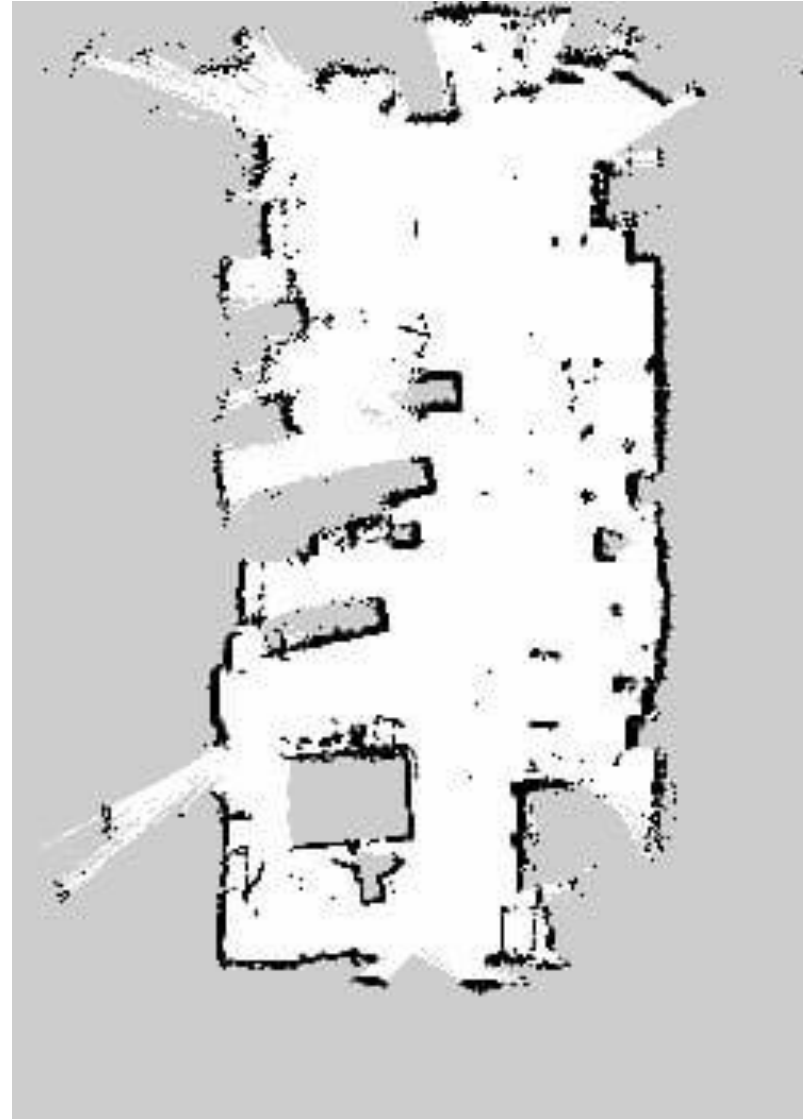


Figure 16: Created Map of Robotics Club workspace using SLAMToolbox Plugin



AMR Navigation using Nav2



RViz*

File Edit View Help

Move Camera Select Focus Camera Measure 2D Pose Estimate 2D Goal Pose Publish Point + -

Status: Ok

Selected Frame OK

Model

Scan

Status: Ok

Topic /global_costmap/costmap

Costmap Topic /global_costmap/costmap

Resolution 0.7

Scheme map

Behind

Resolution 0.05

Size 316

Height 259

Position -4.4612; -3.5824; 0

Orientation 0; 0; 1

Timestamp

Information

Duplicate Remove Rename

1712577213.79 ROS Elapsed: 1570.53 Wall Time: 1712577213.82 Wall Elapsed: 1570.53

Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click/Mouse Wheel: Zoom. Shift: More options.

Views

Type: Orbit (rviz_defaul) Zero

Current View Orbit (rviz)

Near Clip D... 0.01

Invert Z Axis

Target Fra... <Fixed Frame>

Distance 16.3426

Focal Shap... 0.05

Focal Shap...

Yaw 0.140403

Pitch 1.2598

Focal Point 2.273; 2.9421; -1...

Save Remove Rename

Experimental

31 fps

5:38 PM 4/8/24



Material Handling Design and Fabrication

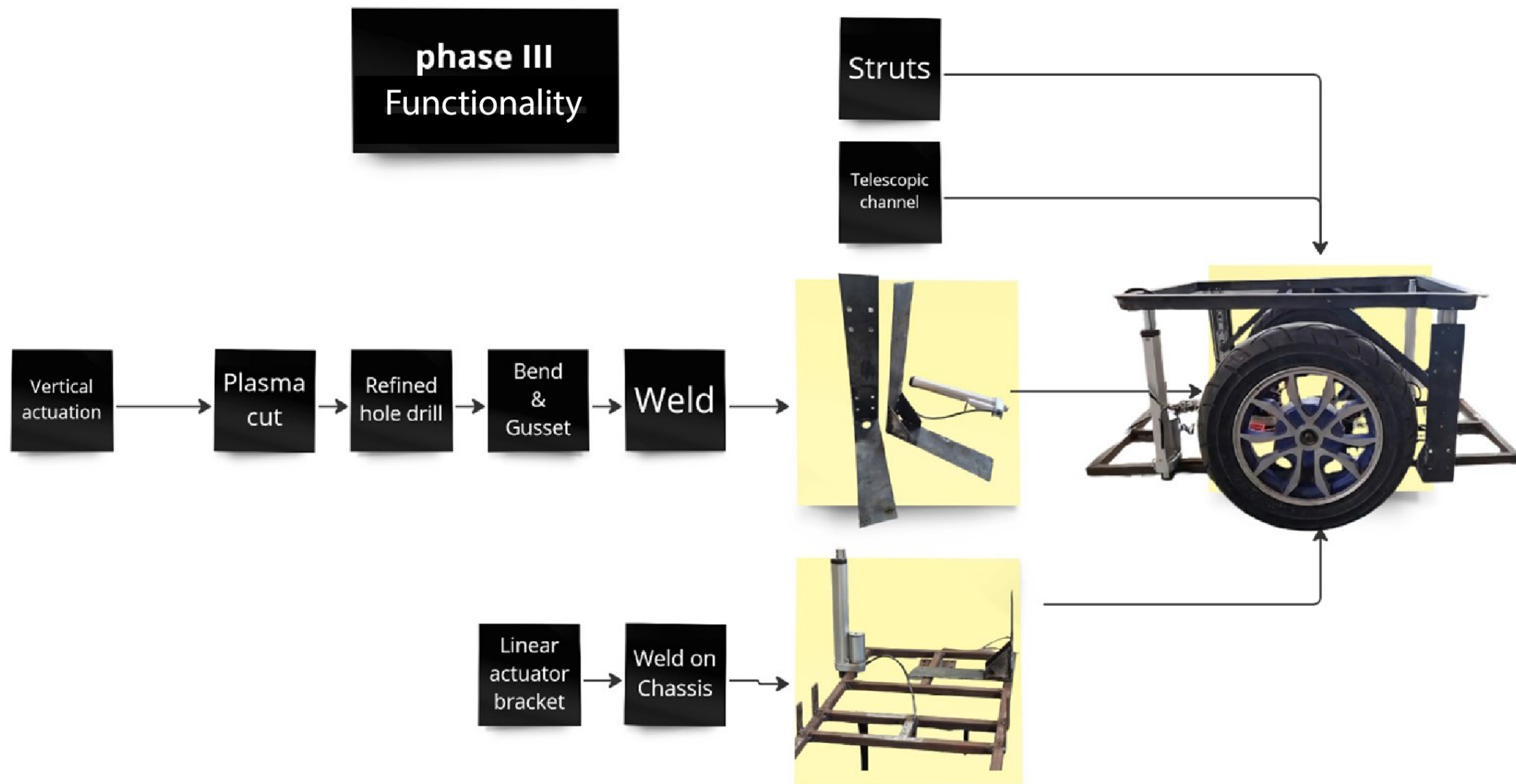


Figure 17 : Fabrication of Material Handling subsystem



Circuit Design and Fabrication

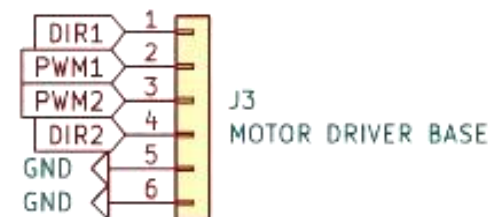
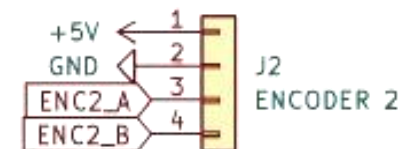
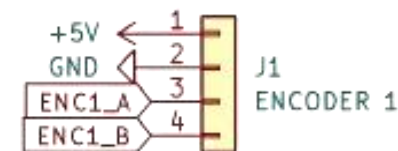
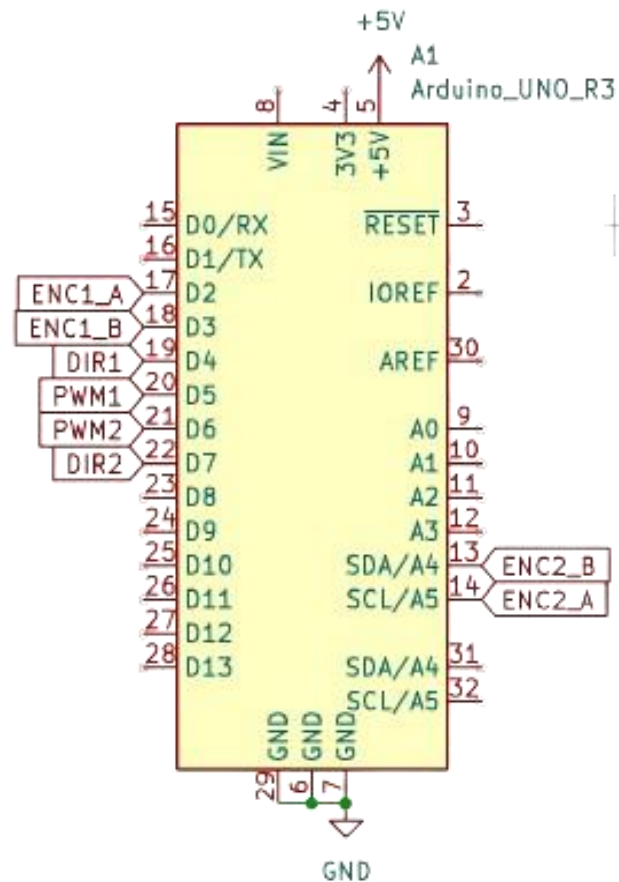
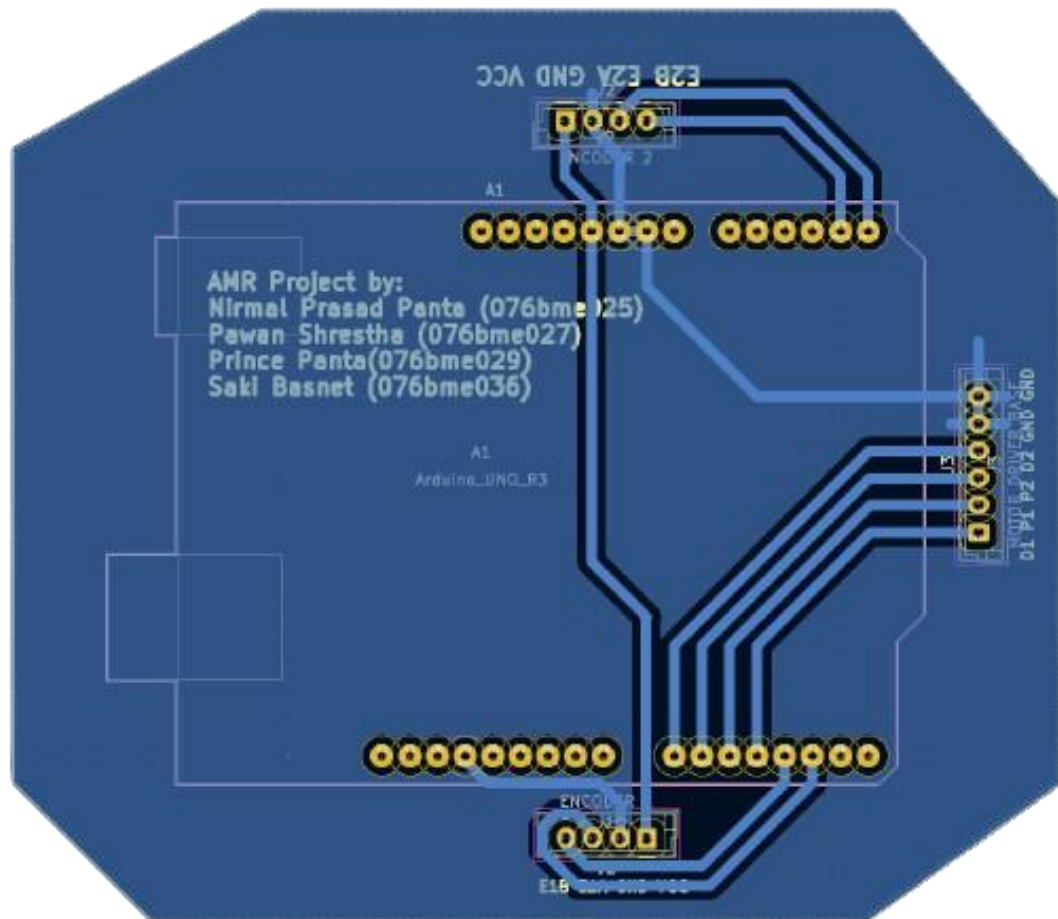


Figure 18: Circuit design in KiCAD

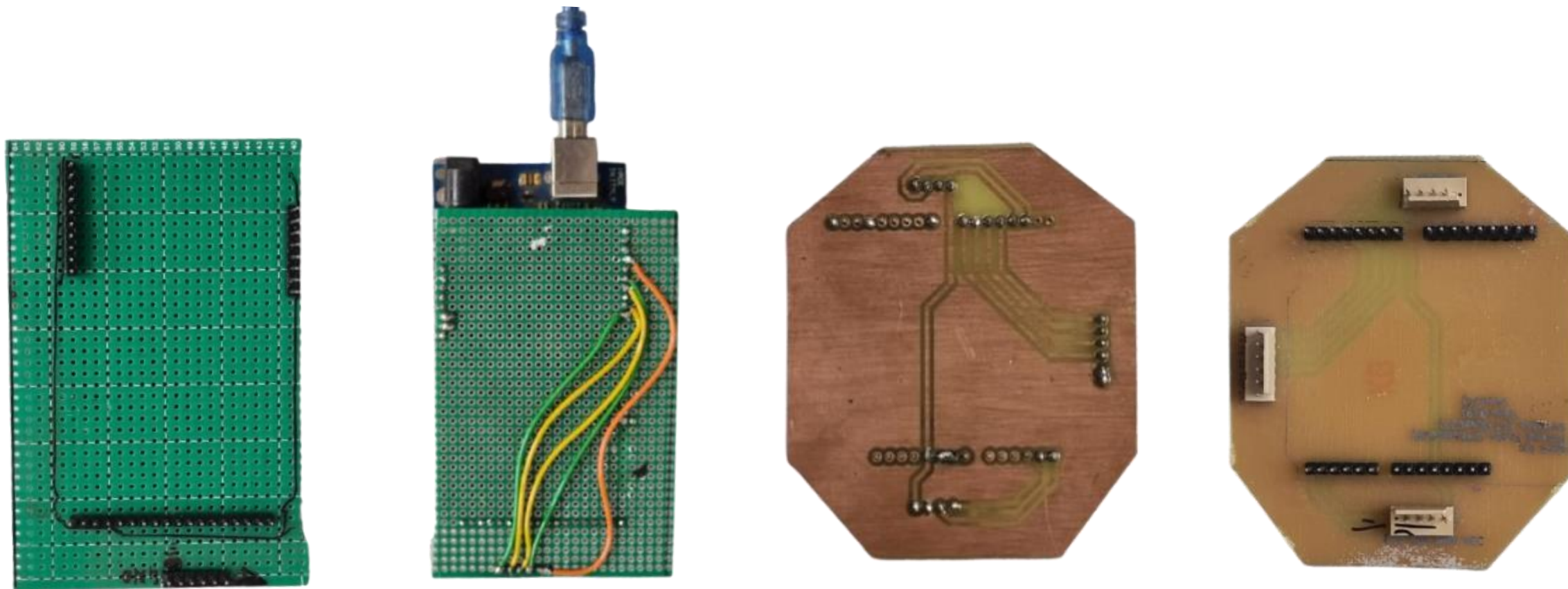


Figure19: Matrix board and fabricated PCB



Teleop Material Handling

(prince) 192.168.164.5 — Konsole

```
File Edit View Bookmarks Plugins Settings Help
[INFO] [launch]: Default logging verbosity is set to INFO
[INFO] [robot_state_publisher-1]: process started with pid [117445]
[INFO] [twist_mux-2]: process started with pid [117448]
[robot_state_publisher-1] [INFO] [1712222490.381237124] [robot_state_publisher]: got segment base_footprint
[robot_state_publisher-1] [INFO] [1712222490.382466847] [robot_state_publisher]: got segment base_link
[robot_state_publisher-1] [INFO] [1712222490.382630032] [robot_state_publisher]: got segment camera_link
[robot_state_publisher-1] [INFO] [1712222490.382680217] [robot_state_publisher]: got segment camera_link_optical
[robot_state_publisher-1] [INFO] [1712222490.382713365] [robot_state_publisher]: got segment caster_wheel_back
[robot_state_publisher-1] [INFO] [1712222490.382740865] [robot_state_publisher]: got segment caster_wheel_front
[robot_state_publisher-1] [INFO] [1712222490.382767939] [robot_state_publisher]: got segment chassis
[robot_state_publisher-1] [INFO] [1712222490.382798921] [robot_state_publisher]: got segment laser_frame
[robot_state_publisher-1] [INFO] [1712222490.382826791] [robot_state_publisher]: got segment left_wheel
[robot_state_publisher-1] [INFO] [1712222490.382853458] [robot_state_publisher]: got segment right_wheel
[twist_mux-2] [INFO] [1712222490.391825087] [twist_mux]: Topic handler 'topics.joystick' subscribed to topic 'cmd_vel_joy': timeout = 0.500000s, priority = 100.
[twist_mux-2] [INFO] [1712222490.430690550] [twist_mux]: Topic handler 'topics.navigation' subscribed to topic 'cmd_vel': timeout = 0.500000s, priority = 10.
[twist_mux-2] [INFO] [1712222490.435013087] [twist_mux]: Topic handler 'topics.tracker' subscribed to topic 'cmd_vel_tracker': timeout = 0.500000s, priority = 20.
[INFO] [ros2_control_node-3]: process started with pid [117757]
[INFO] [spawner-4]: process started with pid [117759]
[INFO] [spawner-5]: process started with pid [117761]
[ros2_control_node-3] [WARN] [1712222497.643133416] [controller_manager]: [Deprecated] Passing the robot description parameter directly to the control_manager node is deprecated. Use '~/robot_description' topic from 'robot_state_publisher' instead.
[ros2_control_node-3] [INFO] [1712222497.646742250] [resource_manager]: Loading hardware 'RealRobot'
[ros2_control_node-3] [INFO] [1712222497.651833176] [resource_manager]: Initialize hardware 'RealRobot'
[ros2_control_node-3] [INFO] [1712222497.655643861] [DiffDriveArduino]: Configuring...
[ros2_control_node-3] [INFO] [1712222497.662714861] [DiffDriveArduino]: Finished Configuration
[ros2_control_node-3] [INFO] [1712222497.666368287] [resource_manager]: Successful initialization of hardware 'RealRobot'
[ros2_control_node-3] [INFO] [1712222497.667133324] [resource_manager]: 'configure' hardware 'RealRobot'
[ros2_control_node-3] [INFO] [1712222497.667292398] [resource_manager]: Successful 'configure' of hardware 'RealRobot'
[ros2_control_node-3] [INFO] [1712222497.667340750] [resource_manager]: 'activate' hardware 'RealRobot'
[ros2_control_node-3] [INFO] [1712222497.667367213] [DiffDriveArduino]: Starting Controller...
[ros2_control_node-3] [INFO] [1712222499.671410582] [resource_manager]: Successful 'activate' of hardware 'RealRobot'
[ros2_control_node-3] [INFO] [1712222500.210656748] [controller_manager]: update rate is 30 Hz
[ros2_control_node-3] [INFO] [1712222500.283877637] [controller_manager]: RT kernel is recommended for better performance
[ros2_control_node-3] [INFO] [1712222500.732102081] [controller_manager]: Loading controller 'joint_broad'
[spawner-4] [INFO] [1712222500.898269581] [spawner_joint_broad]: Loaded joint_broad
[ros2_control_node-3] [INFO] [1712222500.924250414] [controller_manager]: Configuring controller 'joint_broad'
[ros2_control_node-3] [INFO] [1712222500.924711822] [joint_broad]: 'joints' or 'interfaces' parameter is empty. All available state interfaces will be published
[ros2_control_node-3] [INFO] [1712222501.037201970] [controller_manager]: Loading controller 'diff_cont'
[spawner-4] [INFO] [1712222501.052793988] [spawner_joint_broad]: Configured and activated joint_broad
[spawner-5] [INFO] [1712222501.227953562] [spawner_diff_cont]: Loaded diff_cont
[ros2_control_node-3] [INFO] [1712222501.267646599] [controller_manager]: Configuring controller 'diff_cont'
[spawner-5] [INFO] [1712222501.465522470] [spawner_diff_cont]: Configured and activated diff_cont
[INFO] [spawner-4]: process has finished cleanly [pid 117759]
[INFO] [spawner-5]: process has finished cleanly [pid 117761]
```

(prince) 192.168.164.5 x (prince) 192.168.164.5 x (prince) 192.168.164.5 x

rqt_image_view_ImageView - rqt


Image View

/image_raw/compressed

0 10.00m

/image_raw/compressed_mouse_left

Smooth scaling 0°





Teleop Material Handling

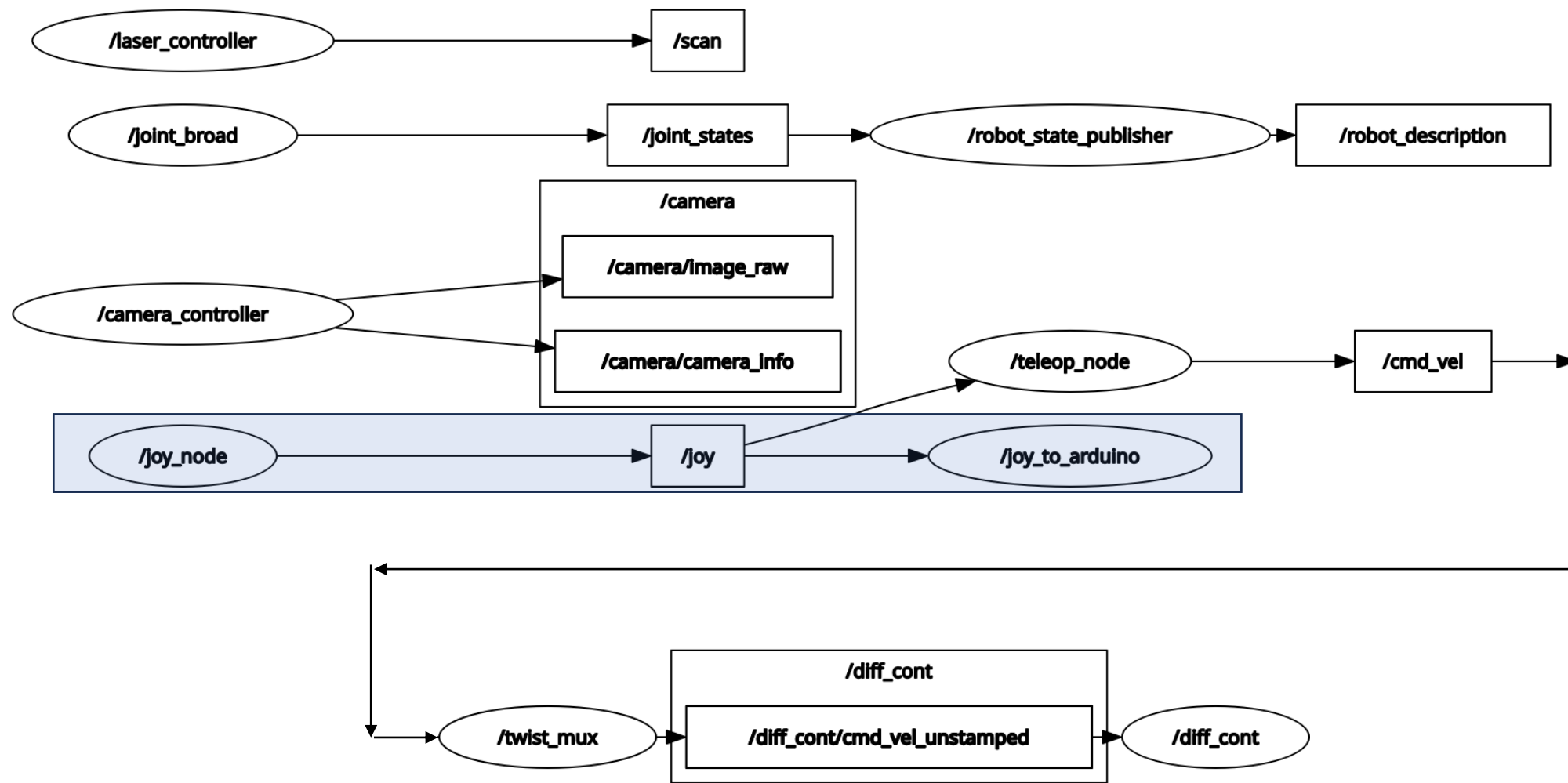
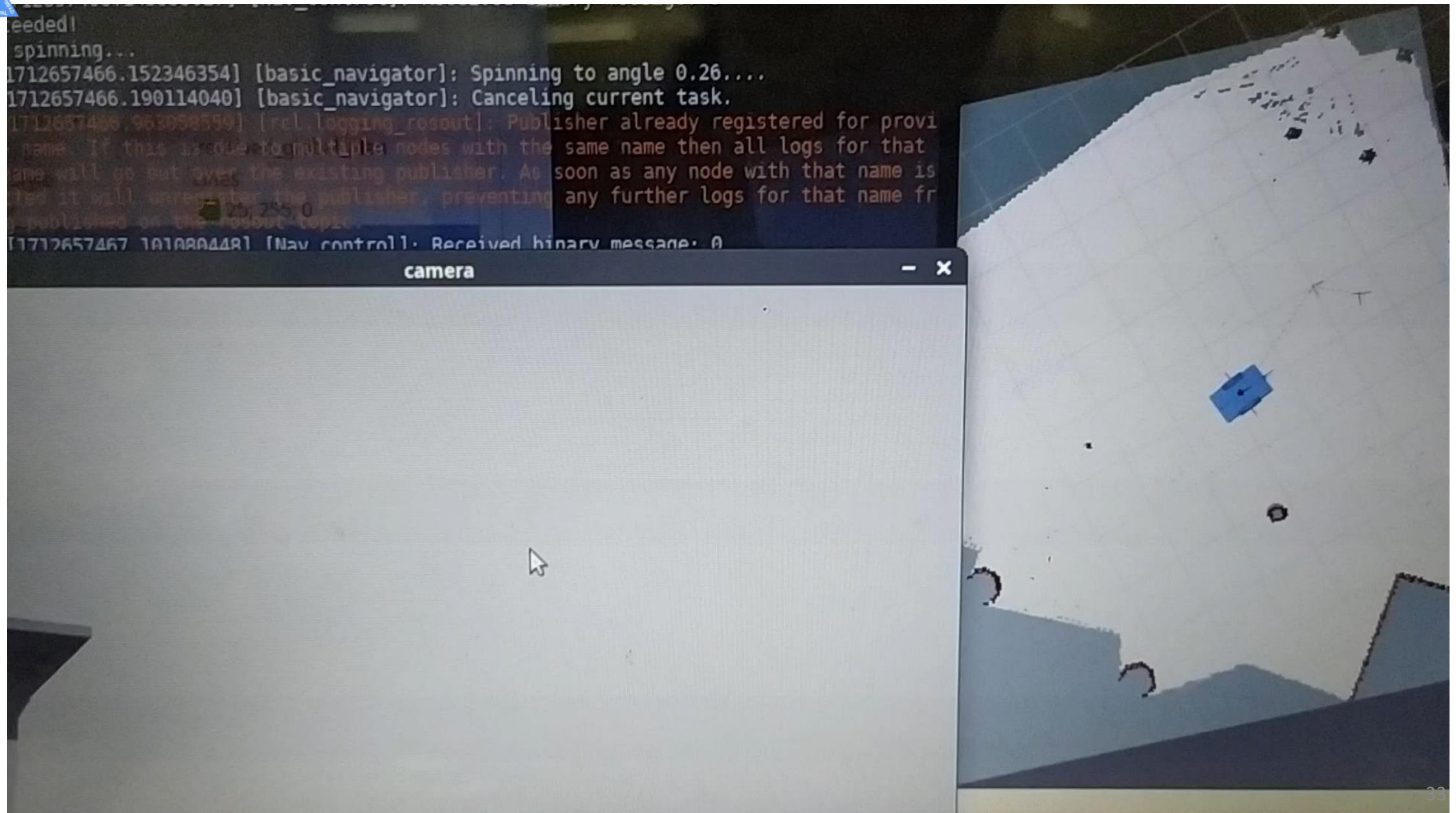


Figure 20: RQT graph during Teleop Material Handling



Autonomous Material Handling Simulation





AMR Control Architecture

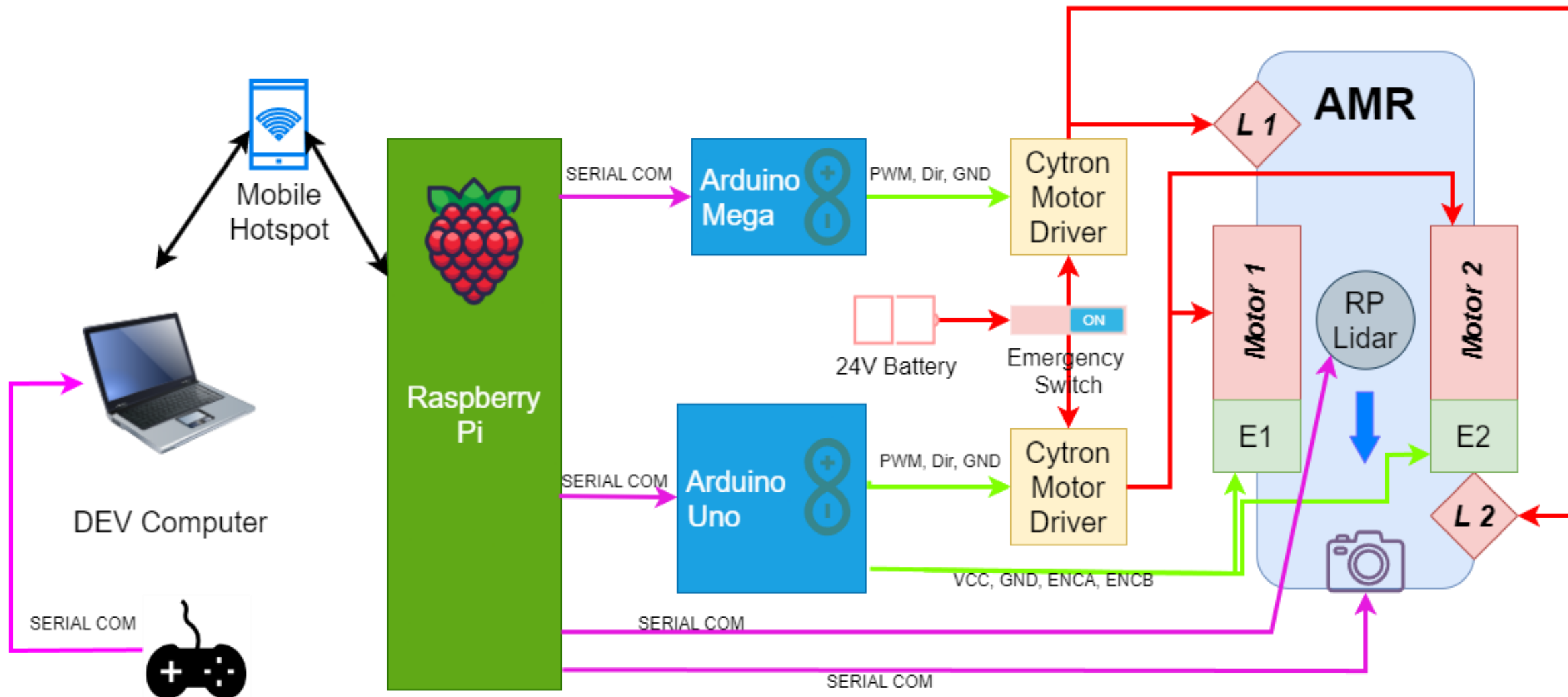


Figure 21: Final AMR Control Architecture



AMR Control Architecture

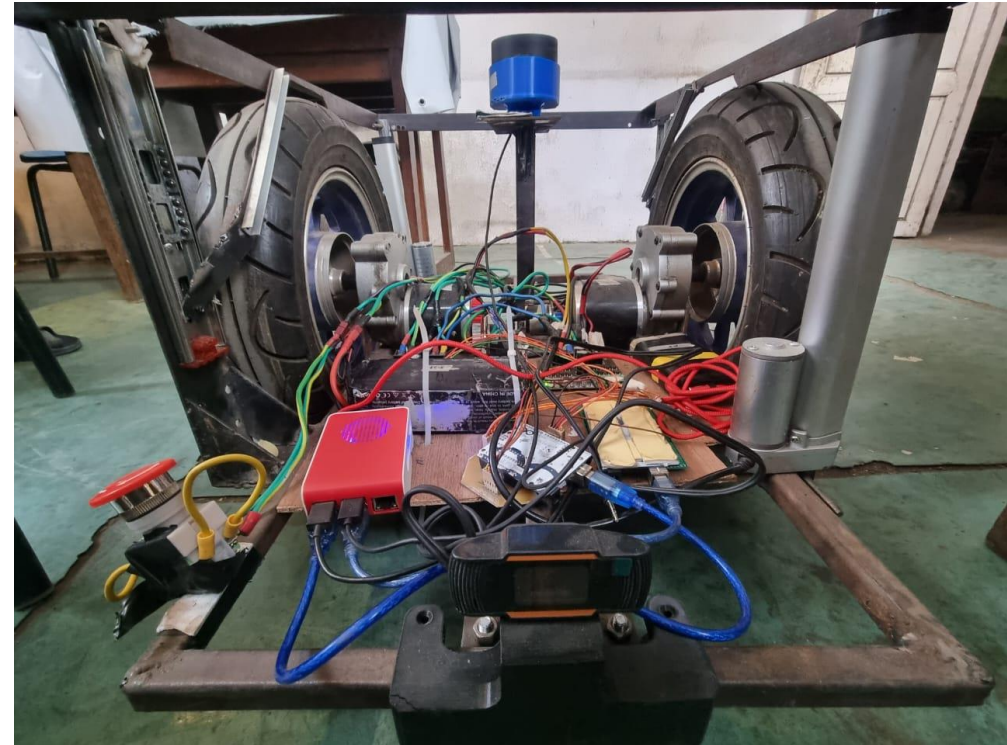
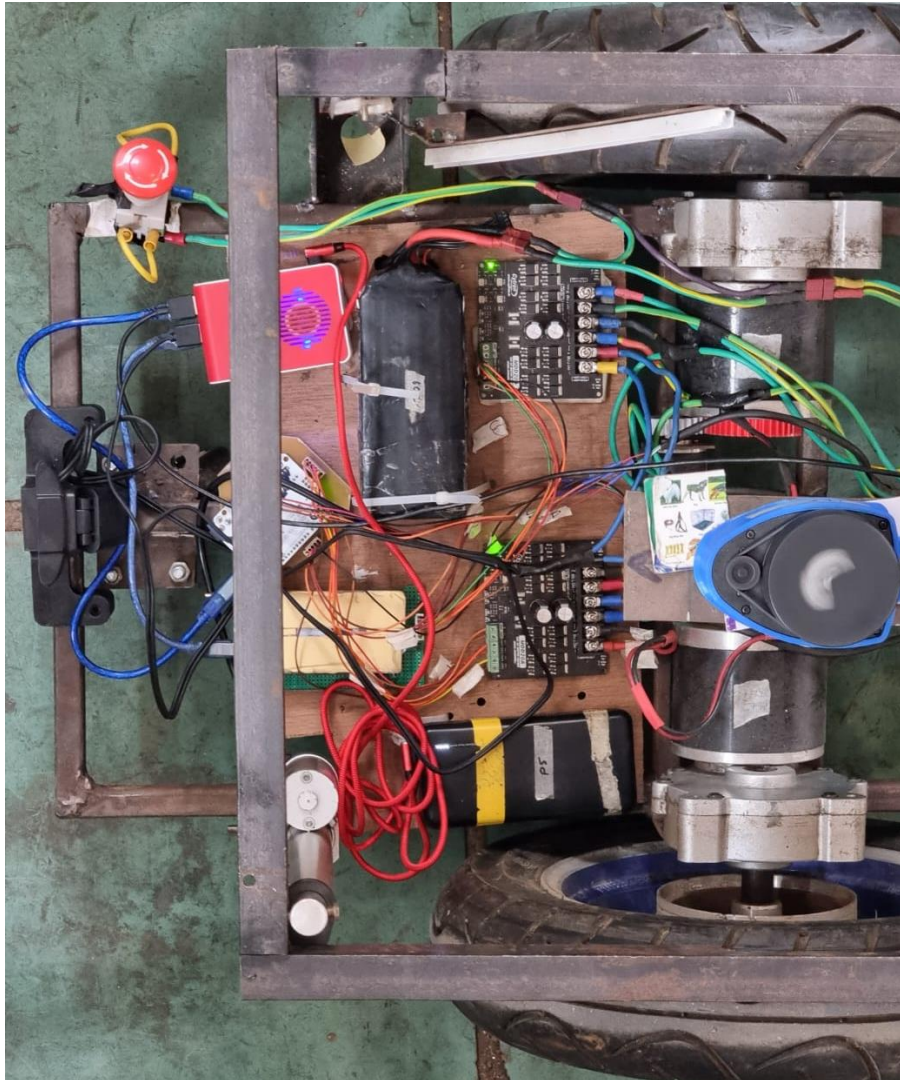


Figure 22 : Actual Control Circuit



OUTPUT: Autonomous Material Handling

RViz

File Panels Help

Interact Move Camera Select Focus Camera Measure 2D Pose Estimate 2D Goal Pose Publish Point

Displays

- Global Options
 - Fixed Frame: map
 - Background Color: 48; 48; 48
 - Frame Rate: 30
- Global Status: Ok
- Fixed Frame: OK
- Grid:
- TF:
- RobotModel:
- LaserScan:
- Map:
- Map:
- Path:

Topic

nav_msgs/msg/Path topic to subscribe to.

Add Duplicate Remove Rename

Time

ROS Time: 1712671134.58 ROS Elapsed: 1200.47 Wall Time: 1712671134.61 Wall Elapsed: 1200.47

Reset Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click/Mouse Wheel: Zoom. Shift: More options.

final_ws : bash — Konsole

File Edit View Bookmarks Plugins Settings Help

```
Estimated time of arrival: 0.0 seconds.
Estimated time of arrival: 0.0 seconds.
Estimated time of arrival: 0.0 seconds.
Estimated time of arrival: 0.0 seconds.
Estimated time of arrival: 0.0 seconds.
Estimated time of arrival: 0.0 seconds.
Estimated time of arrival: 0.0 seconds.
Estimated time of arrival: 0.0 seconds.
Estimated time of arrival: 0.0 seconds.
Estimated time of arrival: 0.0 seconds.
Estimated time of arrival: 0.0 seconds.
^CTraceback (most recent call last):
  File "/home/princess/final_ws/install/robot_control/lib/robot_control/demo_pick_final", line 33, in <module>
    sys.exit(load_entry_point('robot-control', 'console_scripts', 'demo_pick_final')())
  File "/home/princess/final_ws/build/robot_control/robot_control/demo_pick_final.py", line 115, in main
    rclpy.spin(node)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/__init__.py", line 222, in spin
    executor.spin_once()
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 739, in spin_once
    self.spin_once_impl(timeout_sec)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 734, in _spin_once_impl
    handler()
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/task.py", line 239, in __call__
    self._handler.send(None)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 437, in handler
    await call_coroutine(entity, arg)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 362, in _execute_subscription
    await await_or_execute(sub.callback, msg)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 107, in await_or_execute
    return callback(*args)
  File "/home/princess/final_ws/build/robot_control/robot_control/demo_pick_final.py", line 78, in navigation
    while not navigator.isTaskComplete():
  File "/opt/ros/humble/lib/python3.10/site-packages/nav2_simple_commander/robot_navigator.py", line 283, in isTaskComplete
    rclpy.spin_until_future_complete(self, self.result_future, timeout_sec=0.10)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/__init__.py", line 248, in spin_until_future_complete
    executor.spin_until_future_complete(future, timeout_sec)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 310, in spin_until_future_complete
    self.spin_once_until_future_complete(future, timeout_left)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 746, in spin_once_until_future_complete
    self.spin_once_impl(timeout_sec)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 728, in _spin_once_impl
    handler, entity, node = self.wait_for_ready_callbacks(timeout_sec=timeout_sec)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 711, in wait_for_ready_callbacks
    return next(self._cb_iter)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 608, in _wait_for_ready_callbacks
    wait_set.wait(timeout_nsec)
KeyboardInterrupt
[ros2run]: Interrupt
princess@princess:~/final_ws$
```

...s: ros2 x ...s: ros2 x ...s: ros2 x ...s: ros2 x ... ros2 x ... ros2 x ..._bash x ...mplescreenrec x ...viz2 x ...bash x

7:43 PM 4/9/24



Problems Faced

- Limited resources so if a problem occurred in a hardware, work would halt
- Failure in debugging errors which forced to start over
- Visual manager of Ubuntu for Raspberry PI not functioning
- Error in camera calibration using ArUco tag – discrepancy in distance obtained using CV and actual distance
- Major issue faced- Network Latency and Pi Crash
 - Localization delayed and as a result robot gets stalled or worse hits the obstacle
 - Real time video frame published lags and as a result not feasible in using camera feed if problem in navigation in remote location



Limitations

- The size of the AMR is constrained due to the use of available motors
- Different environments not considered for the operation of LIDAR
- Fleet of AMRs not considered which is the norm in the industry
- AMR is not capable of detecting obstacles below the lidar level
- Linear actuators operate in open loop
- The accuracy and precision of navigating and docking of AMR is affected due to network latency and camera calibration error respectively



Working Schedule

Gantt Chart

20-May 30-Jun 10-Aug 20-Sep 31-Oct 11-Dec 21-Jan 2-Mar 10-Apr

Literature Review

271

Proposal

11

AMR Design

193

AMR Fabrication

209

ROS2 setup and Simulation

183

AMR Control

85

SLAM and Nav2 Setup

44

AMR test with SLAM and Nav2

41

CV setup

58

AMR test with CV

60

Documentation

127

Phase I

Phase II

Phase III

Start Date Days Completed

Figure 23: Project Gantt Chart



Budget analysis

Table 1: Base Cost

S.N.	Description	Estimation in NRs
1	Manufacturing Cost	20,000
2	Logitech C270 Camera	3,895
3	Documentation	5,000
4	Miscellaneous	2,000
5	Raspberry Pi 4	20,000
Total		50,895

Table 2: Items to be pledged

S.N.	Description	Estimation in NRs
1	Motor Drivers	3,000
2	Hub Motors	22,000
3	Arduino	2,380
4	LiDAR	15,000
5	Linear Actuator	8,000
Total		50,380



CONCLUSION AND FUTURE ENHANCEMENT

AMR developed and tested for material handling successfully in three phases:

Phase I :
Transport

- AMR Chassis Design and Fabrication
- ROS2 Setup and Simulation
- ROS control integration and Teleoperation of AMR

Phase II :
Improvement
on Transport

- Integration of SLAM for mapping and Nav2 for navigation
- Testing of AMR for autonomous mobility

Phase III :
Functionality
to Transport

- AMR material handling design and fabrication
- Teleoperated material handling
- CV setup using ArUco tag detection and camera calibration
- Autonomous navigation, docking and material handling



Conclusion and Future enhancements

- Minimizing height of the robot using low radius wheel and high torque motor or use of ultrasonic sensors for low obstacle detection
- System incorporating fleets of robot can be developed and path and task planning optimization such as Dijkstra's algorithm can be implemented.
- Used of additional sensors like Inertial Measuring Unit (IMU) can be incorporated to enhance the odometry data.
- Improvement upon existing design such as round body instead of rectangular and suspension for caster wheels
- Use of Visual SLAM with cameras for improved pose estimation and 3D mapping



References

- [1] G. Ullrich, "The History of Automated Guided Vehicle Systems," in Automated Guided Vehicle Systems: A Primer with Practical Applications, Berlin, Heidelberg, Springer Berlin Heidelberg, 2015, pp. 1-14.
- [2] W. Grzechca, "Manufacturing in Flow Shop and Assembly Line Structure," International Journal of Materials, Mechanics and Manufacturing, vol. 4, no. 1, pp. 25-30, 2016.
- [3] N. Zghair, A. Khaleq and A. S. Al-Araji, "A one decade survey of autonomous mobile robot systems," International Journal of Electrical and Computer Engineering, vol. 11, no. 6, p. 4891, 2021.
- [4] L. Lynch, T. Newe, J. Clifford, J. Coleman, J. Walsh and D. Toal, "Automated Ground Vehicle (AGV) and Sensor Technologies- A Review," 2018 12th International Conference on Sensing Technology (ICST), Limerick, Ireland, 2018, pp. 347-352, doi: 10.1109/ICSensT.2018.8603640.
- [5] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, W. Woodall, "Robot Operating System 2: Design, architecture, and uses in the wild," Science Robotics vol. 7, May 2022.
- [6] Macenski et al., (2021). SLAM Toolbox: SLAM for the dynamic world. Journal of Open Source Software, 6(61), 2783. <https://doi.org/10.321105/joss.02783>
- [7] G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, 2000.
- [8] J. Wubben, F. Fabra, C. T. Calafate, T. Krzeszowski, J. M. Marquez-Barja, J.-C. Cano and P. Manzoni, "Accurate Landing of Unmanned Aerial Vehicles Using Ground Pattern Recognition," Electronics, vol. 8, 2019.



Acknowledgements

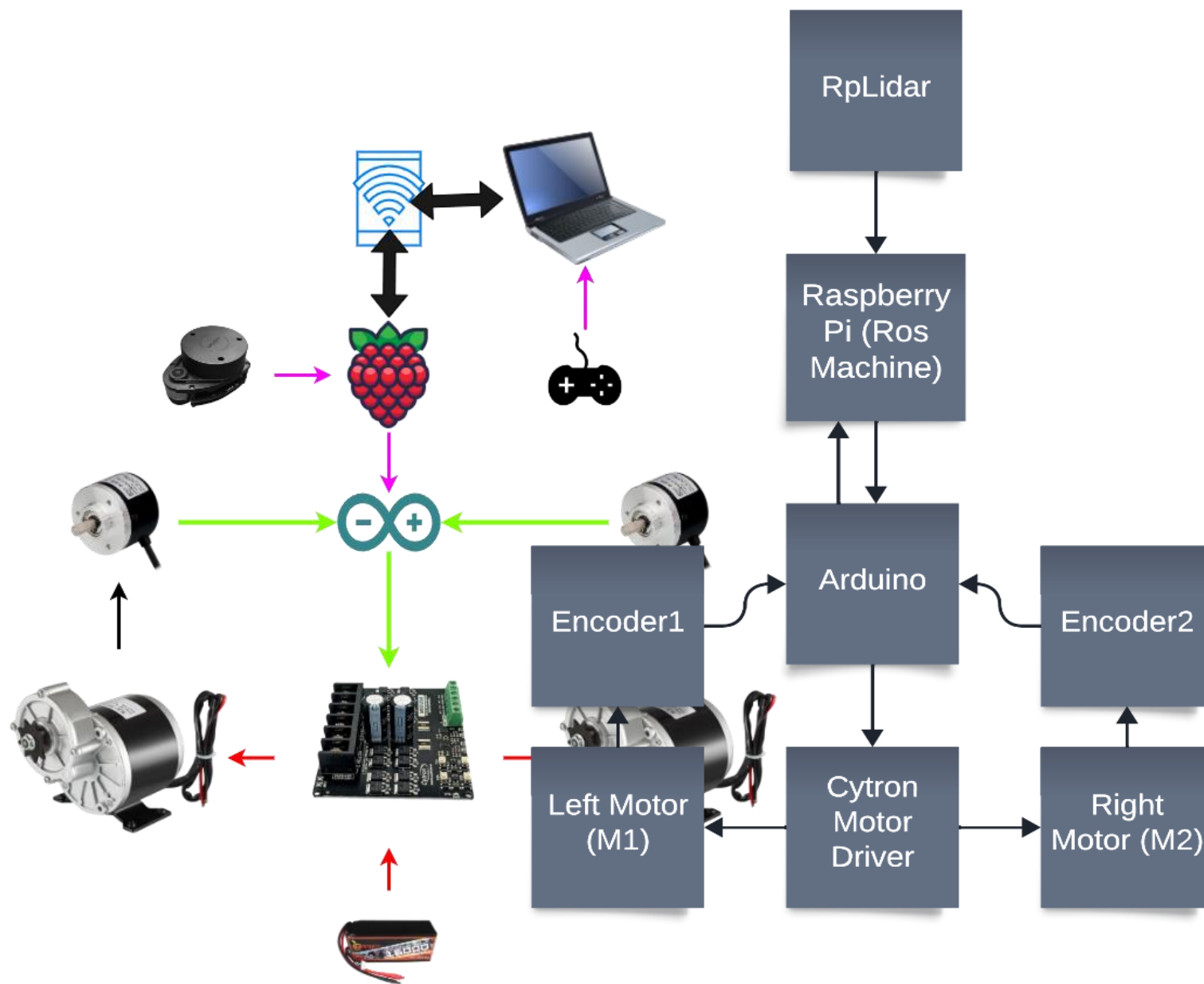
- Department of Mechanical and Aerospace Engineering, Pulchowk Campus and HOD Dr. Sudip Bhattarai
- Project Supervisor, Associate Prof. Dr. Surya Prasad Adhikari
- Deputy Head of Department and Assistant Professor Laxman Motra
- Deputy Head of Department and Assistant Professor Kamal Darlami
- Assistant Professor Aashish Karki
- Robotics Club, Pulchowk Campus; especially Ramesh Chaudhary sir, Alumni Nitesh Subedi and Mr. Sabin Shrestha
- To all the teachers, friends and seniors and juniors who have assisted us directly or indirectly in this project



THANK YOU!



Teleoperated AMR Locomotion Control





Creation of Robot Model

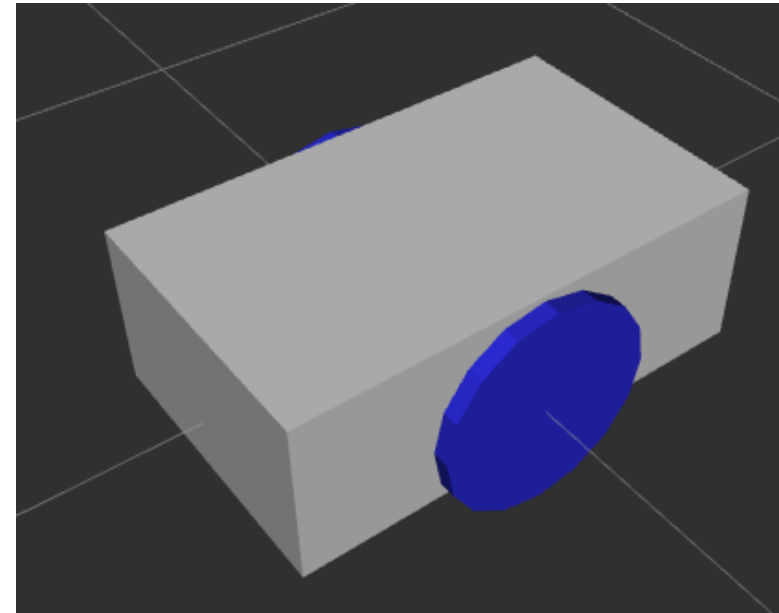
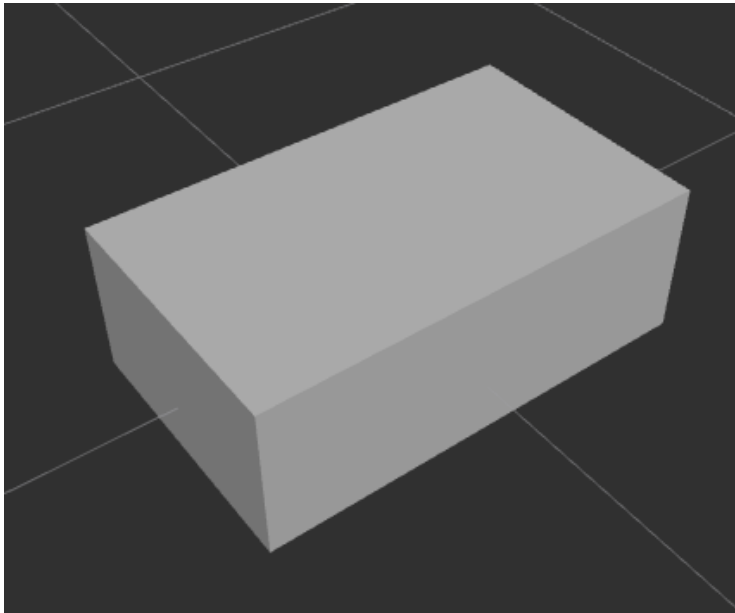


Figure 10 : Creation of simplified robot model

NAV2

