CG2111A Engineering Principle and Practice II
Semester 2 2023/2024

"Alex to the Rescue"
Design Report
Team: B04-4B

NAME	STUDENT	SUB-TEAM	ROLE
MARK NEO QI HAO	A0281484X	Hardware	Circuitry
POH YU WEN	A0271715E	Software	Lidar
SAW SHENG JIE, ISAAC	A0272400W	Firmware	Motor control
NGUYEN DUC PHONG	A0276035E	Software	Serial comms

Section 1 System Functionalities

The Alex robot is a sophisticated system that integrates an Arduino Uno and a Raspberry Pi 4, complemented by four motors and a LIDAR system mounted on its chassis. Control over the Alex robot is facilitated through laptop commands transmitted via WIFI, utilizing a Virtual Network Computing (VNC) server. This server serves as the conduit for data exchange, managed by the Raspberry Pi, which subsequently relays these commands to both the Arduino and the LIDAR system. This orchestration allows the robot to execute movements and gather environmental data effectively.

The primary objective of the Alex robot is to navigate and map its surroundings using the precise data obtained from the LIDAR system. It is designed to accomplish this task with finesse, culminating in successful maneuvering and positioning within a designated 1 by 1 "parking" spot in a maze-like environment. To enhance its capabilities further, a color sensor has been incorporated to identify specific objects of interest, such as red and green markers, mimicking scenarios where it may need to locate and assist individuals, akin to earthquake victims.

Additionally, to provide intuitive control over the robot's movements, a PlayStation 4 controller has been integrated into the system, enabling seamless interaction and precise maneuvering commands sent from the laptop interface.

Section 2 Review of State of the Art

Boston Dynamics Spot Robot

This system integrates advanced hardware and software components for tele-operated search and rescue missions. It typically includes a rugged chassis equipped with high-resolution cameras, LIDAR sensors, and manipulator arms. The software component comprises real-time communication protocols, robust navigation algorithms, and intuitive user interfaces.

[1]

Strengths

- High-resolution cameras and LIDAR sensors provide detailed environmental data for accurate mapping and navigation.
- Real-time communication and navigation algorithms enable responsive control and efficient maneuvering in dynamic scenarios.

Weaknesses

- Dependence on tele-operation limits autonomy, especially in communication-challenged environments.

DJI Matrice 300 RTK

This system emphasizes mobility and adaptability in search and rescue missions. It typically features a modular design with interchangeable sensors and payloads, supported by a robust communication network. The hardware includes ruggedized wheels or tracks, along with a variety of sensors such as cameras, thermal imaging devices, and gas detectors. Software components encompass intelligent navigation algorithms, machine learning for object recognition, and tele-operation interfaces.

[2]

Strengths

- Diverse sensor suite provides comprehensive situational awareness for effective search and rescue operations.
- Intelligent navigation algorithms optimize path planning and obstacle avoidance.
- Machine learning capabilities enhance object recognition and autonomous decision-making.

Weaknesses

- Limited tele-operation range in remote or communication-challenged areas.
- Initial setup and calibration of sensors and software modules can be time-consuming.

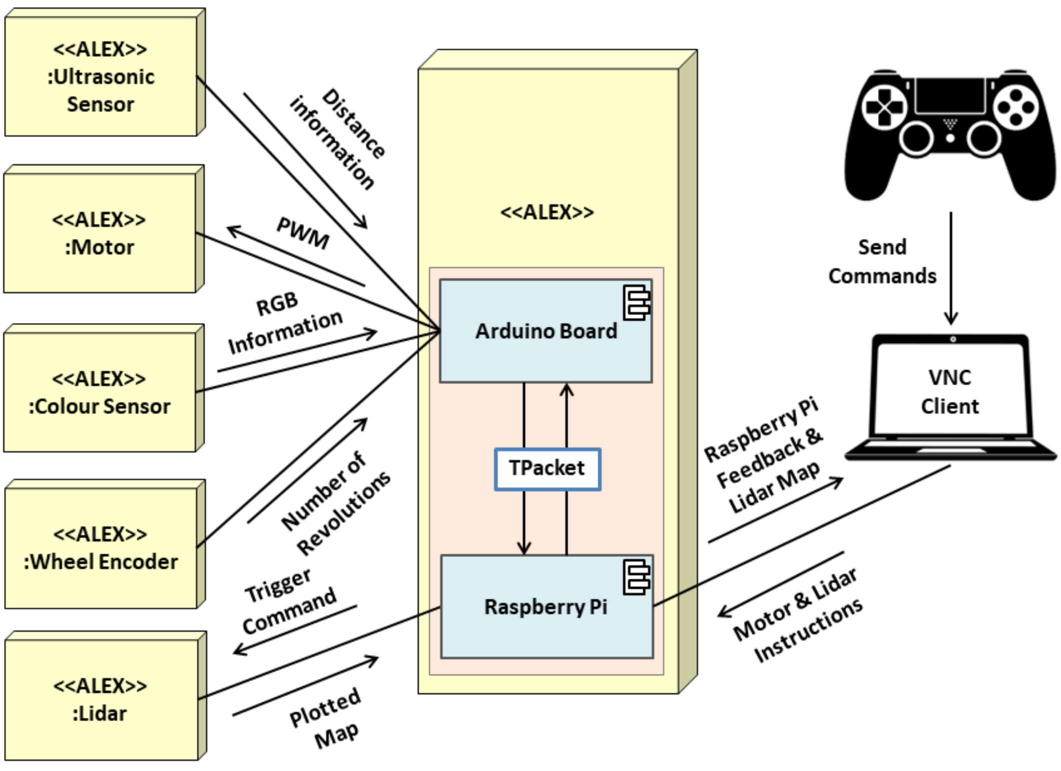
Section 3 System Architecture

The Alex robot is an intricate system comprising essential components such as a wheel encoder, Lidar, color sensor, Arduino Uno, Raspberry Pi 4, and four motors. These components communicate with each other through various channels, including UART, Wi-Fi, and wired connections.

To facilitate movement, commands are initiated via a PS4 controller, which are then transmitted remotely via VNC from a laptop. It is crucial for these components to synchronize seamlessly to ensure accurate processing of commands.

Furthermore, the Lidar readings play a pivotal role in mapping the maze environment. These readings are relayed back to the laptop to provide the user with a comprehensive understanding of the maze layout.

The overall algorithm governing the interaction of these components within the Alex robot is paramount. It orchestrates the collaborative effort of each component, ensuring efficient communication, accurate processing of commands, and effective utilization of sensory data for navigation and mapping purposes.



Overview of interactions between devices in the system architecture

Section 4 Component Design

High-Level Steps:

1. Initialize Alex and establish communication between the laptop and the Raspberry Pi.
2. Start the timer for the 6-minute window.
3. Perform an initial scan of the environment using the LiDAR scanner.
4. Send movement commands to the Arduino through the RPi via the controller and the laptop.
5. Based on received commands, the Arduino will rotate the wheels to navigate Alex through the environment.
6. After each movement, we will perform a scan of the environment.
7. If objects are detected, we will move Alex to the desired range from the object. Then, we will utilize the colour sensor to identify objects and distinguish victims (green or red) from other objects.
8. Throughout the process, Alex will continuously update the operator with the current map and any detected victims.
9. Repeat steps 4-8 until the 6-minute window elapses or all victims are located.
10. End the operation and submit the final environment map.

Expansion of Complicated Steps

Step 1: Initialization of Alex

1. Set up the communication interface on both the laptop and the Raspberry Pi.
2. Ensure that the Raspberry Pi and the laptop are on the same Wi-Fi Network.
3. Pi sends to the laptop a handshake packet to confirm that the connection is successful.
4. Perform status check and calibration for all sensors (colour, ultrasonic, wheel encoders) by ensuring that they are providing proper output.

Step 4: Send movement commands to the Arduino through the RPi via the controller and the laptop.

Hardware Components: Laptop, Raspberry Pi, Arduino, communication interface (Wi-Fi/Mobile Data), Controller.

1. Using the controller, which is connected to the laptop, input movement commands by pressing the buttons on the controller.
2. The Raspberry Pi should send the corresponding pre-mapped commands over to the Arduino via a command packet.
3. Then, the Arduino processes the command packet and executes the command to move Alex.

Step 5: Navigate Alex Through the Environment

Data: Commands for movement and navigation map data.
Hardware Components: Motors for movement control, wheel encoders for distance measurement, Raspberry Pi for processing and decision-making.

1. Use wheel encoders to measure the distance traveled by Alex for accurate movement control.
2. Ensure that the distance recorded by the wheel encoders matches the distance to move from the commands.
3. Continuously monitor sensor data from the ultrasonic sensor to detect obstacles and manually adjust the navigation path as needed.
4. When an obstacle is detected, perform a scan of the environment using the LiDAR scanners.
5. Ensure that the orientation of Alex is facing the object by referring to the LiDAR scan and sending the necessary turning commands.
6. As there will be an ultrasonic sensor placed above the colour sensor, we can use the ultrasonic sensor to ensure that the distance from the object is in the desired range before performing the detect object algorithm.

Step 7: Identify Objects and Distinguish Victims

Data: Sensor readings, predefined victim identification criteria.
Hardware Components: Color sensor for detecting object colors, microcontroller for processing sensor data, Raspberry Pi for coordinating object identification algorithms.

1. When there are possible objects detected, the identify objects algorithm will be run.
2. Use the colour sensor to distinguish between victims that are green or red. Then, mark those manually on the map.
3. If the sensor returns a colour other than green or red, classify it as other objects and mark it on the map.
4. Implement error handling mechanisms to address any sensor inaccuracies or environmental variations.

Step 8: Continuously update the operator on the victim and map

Data: Victim locations, environment map.
Hardware Components: Raspberry Pi for updating the environment map visualization, LiDAR.

1. Using ROS, transfer the relevant scanned data from the LiDAR through the Raspberry Pi back to the laptop.
2. Then, create an updated environment map with the manually marked recorded victim locations.

Section 5 Project Plan

Week	Milestone	Deliverables
9	System Setup and Communication	• Establish communication channels (UART, Wi-Fi) between components. • Develop initial code for communication and integration testing.
10	Navigation and Control	• Implement code for movement control using motors and encoders. • Integrate PS4 controller for remote command input.
11	Sensor Integration and Mapping	• Install and calibrate LIDAR, color sensor. • Algorithms for color decoding. • Integrate sensor data with navigation system for mapping.
12	Testing, Refinement, and Documentation	• Conduct comprehensive testing of all functionalities. • Refine algorithms and code for improved performance. • Document system architecture, component design, and usage instructions.
13	Trial and Final Run	• Refinement of Alex based on trial runs

References

[1] Goode, L. (2020, October 26). *Boston Dynamics' Robots won't take our jobs ... yet*. Boston Dynamics' Robots Won't Take Our Jobs … Yet. https://www.wired.com/story/get-wired-podcast-14-boston-dynamics/

[2] Avetics. (n.d.-a). *DJI Enterprise Store*. https://www.avetics.com/dji-enterprise-store/dji-m300