Your Name: Mohamed Amine Boughou
Your REDID: 130624660
Your Teammates names :Samaa Altoaibi
Robot ID:18

Robot Final Project Report

1.1 Problem Statement

Our robot is designed to autonomously navigate a labyrinth environment, using a combination of sonar and line sensors to detect obstacles, identify trash, and return to its starting position. It can make directional decisions dynamically, adjust its path in real-time, and use color recognition to differentiate between navigable paths and trash.

1.2 Sensors

The robot is equipped with two primary types of sensors: line sensors and a sonar sensor. Line sensors are mounted beneath the robot and are used to detect color variations on the ground, allowing the robot to recognize trash marked with black lines. These sensors also help differentiate between walkable paths and trash locations.
The sonar sensor is positioned at the front of the robot and is responsible for obstacle detection. It scans the surrounding environment, providing real-time distance measurements. This enables the robot to avoid walls, detect open paths, and maintain safe distances from obstacles.

1.3 Behaviors, Automation, and Control Architecture

Our robot operates using a state-based architecture, which ensures that it can adapt to various scenarios in the maze. It has the following key states:
• **State 1: PHASE_CALIBRATE** - The robot calibrates its line sensors, ensuring accurate readings. It performs a slow spin to capture the range of sensor values for black and white surfaces.
• **State 2: PHASE_MOVE** - The robot uses its sonar to scan the environment, identifies walkable paths, and makes decisions based on sensor feedback. It prioritizes moving forward if the path is clear, then turns right or left if necessary.
• **State 3: PHASE_DETECT** - The robot uses its line sensors to detect trash. If trash is detected, it stops, spins 360° as a cleaning simulation, and updates its trash count.
• **State 4: PHASE_RETURN** - The robot navigates back to its starting position using the shortest calculated path. It dynamically avoids obstacles while backtracking.
• **State 5: PHASE_STOP** - The robot stops, and the buzzer plays a tone to signal task completion.
Low-level behaviors include servo scanning for obstacle detection, motor control for navigation, and buzzer feedback for user interaction.

1.4 Finite State Automata Diagram

The FSA diagram of our robot illustrates the transitions between its various states. The robot starts in the calibration state, moves to the navigation state, detects trash, returns home, and finally stops with a beeping sound.

Your Name: Mohamed Amine Boughou
Your REDID: 130624660
Your Teammates names :Samaa Altoaibi
Robot ID:18

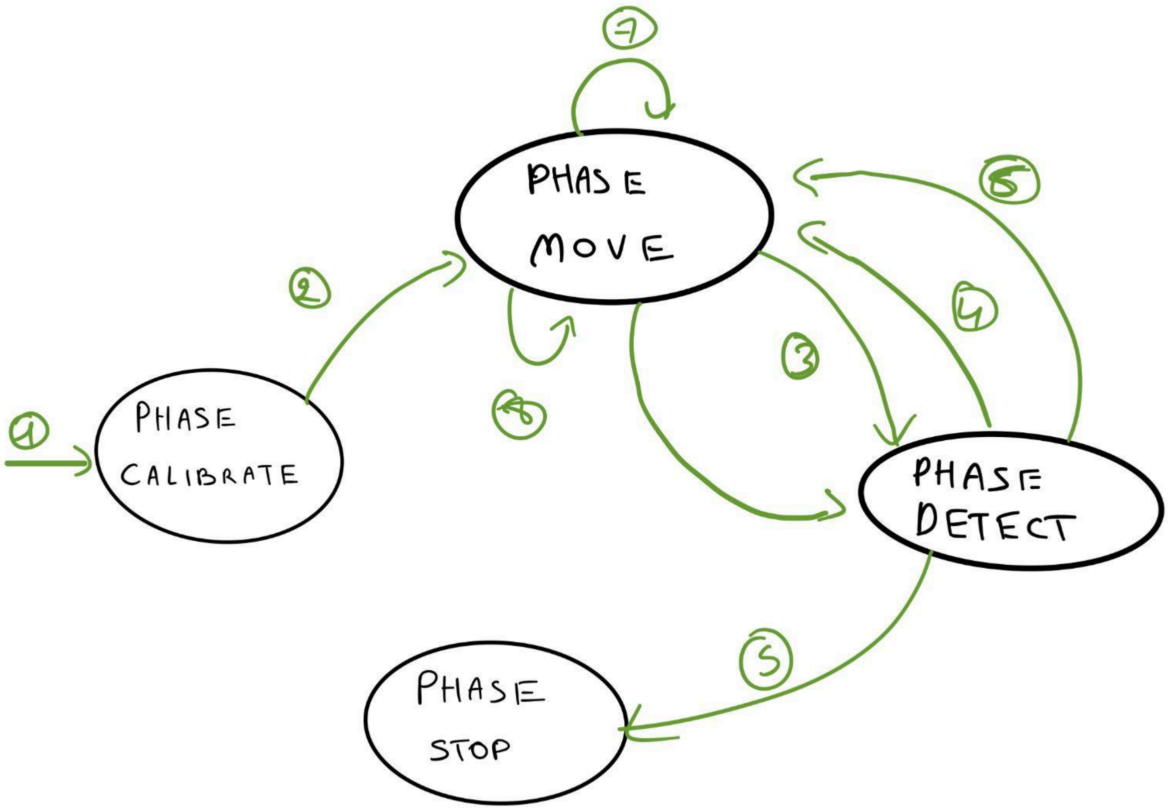
Your Name: Mohamed Amine Boughou
Your REDID: 130624660
Your Teammates names :Samaa Altoaibi
Robot ID:18

1 : After calibration Complete.

2 : After scanning surroundings.

3 : (If trash detected and trash count < 3) ⇒ PHASE MOVE.

4 : (If trash detected and trash count >= 3) ⇒ PHASE STOP.



5 : If no trash detected → [PHASE MOVE]

6 : If obstacles on Forward → [PHASEMOVE] (Turn right on LEFT based on scan)

7 : (If clear Path) → [PHASEMOVE] (move Forward)

8 : (End : Robot stops, buzzer plays, OLED displays status.

Your Name: Mohamed Amine Boughou
Your REDID: 130624660
Your Teammates names :Samaa Altoaibi
Robot ID:18

1.5 Control Architecture

Our robot employs a reactive control architecture. This design allows it to quickly respond to sensor inputs without relying on a pre-programmed map. The control system includes:

• **Threshold-Based Trash Detection**: The line sensors use a fixed threshold to detect trash.

• **Obstacle Avoidance with Sonar**: The sonar sensor continuously scans for walls, and the robot avoids detected obstacles.

• **State-Based Navigation**: The robot follows a state machine that ensures consistent behavior through each phase of operation.

2. Integration of Semester Modules

Our robot's design is directly informed by the coursework:

• **Lab 1**: Provided the foundational concepts of movement and basic motor control.

• **Lab 2**: Introduced sonar-based obstacle detection, which is essential for our robot's navigation.

• **Lab 4**: Implemented state-based automation, forming the backbone of our control logic.

• **Lab 5**: Although wall-following was not fully implemented, the concepts of sensor-based navigation were used for trash detection and obstacle avoidance.

3. Challenges, Error Handling, and Reliability

Challenge 1: Unreliable Trash Identification

Our initial design faced issues with unreliable trash detection. Ambient noise or sensor drift occasionally caused false trash detections or missed trash entirely. This resulted in the robot either failing to collect trash or falsely increasing the trash count.

Error Handling: We addressed this by refining the trash detection threshold, adding a delay after detection to avoid double counting, and adjusting the sensor angle for better surface detection.

Challenge 2: Inconsistent Obstacle Detection

The sonar sensor sometimes misjudged distances, especially on small or angled surfaces. This led to the robot believing that a clear path was blocked or vice versa.

Error Handling: We minimized this problem by recalibrating the sonar sensor, adjusting the detection range, and using servo scanning to cross-check the obstacle data.

Challenge 3: Dynamic Memory on Arduino

Our initial logic actually wasn't eligible for compilation on the program we were using, which was the Arduino, originally, we wanted to use a decision Tree DSA to be able to handle all the cases of visited and unvisited that required 2 arrays, which we implemented a fixed sized array. But verifying the code was a struggle for us due to the dynamic memory. We unfortunately had to give up the initial plan in the last minute to have something deliverable.

4. Testing and Validation

We employed a multi-step testing strategy to ensure the robot performed reliably. Our primary focus was on sensor accuracy, movement precision, and obstacle avoidance.

Calibration Testing

Your Name: Mohamed Amine Boughou
Your REDID: 130624660
Your Teammates names :Samaa Altoaibi
Robot ID:18

We began by testing the line sensors on various surfaces, ensuring they correctly differentiated between trash (black) and the walkable path. Adjustments were made to the threshold values to reduce false detections.

Movement Accuracy

We verified that the robot traveled in straight lines for each tile and that it could make precise 90° turns. Speed and turn duration were adjusted to minimize drift.

Obstacle Detection

We validated that the sonar sensor accurately detected obstacles and allowed the robot to avoid collisions. The servo scanning logic was tested to ensure the robot consistently chose clear paths.

5. Innovation and Creativity

Our robot incorporates several innovative elements that enhanced its functionality beyond basic requirements. The three-way dynamic scanning feature, where the robot looks left, right, and forward before making any movement, allowed it to make more informed decisions and avoid dead ends.

6. Communication and Team Collaboration

Our team maintained consistent and open communication throughout the project. Tasks were distributed based on individual strengths, with some members focusing on sensor integration, others on navigation logic, and some on documentation.
We used regular meetings to discuss progress, troubleshoot issues, and refine our approach. This collaboration ensured a balanced workload and helped maintain momentum, even when challenges arose.

7. Use of Resources

We made extensive use of course materials, including lecture notes on sensor integration, TA feedback for sensor calibration, and online Arduino libraries for efficient coding. Office hours provided direct support for debugging and logic improvements, making them a critical resource. We also explored the Polulu project library to explore different manageable ideas to implement however, it was too late to even thing of anything outside the scope of what we already had due to time struggles.