

# Lego Spike Int. Pro.

*by* Biniyam Memberu

---

**Submission date:** 05-Apr-2025 12:21AM (UTC+0100)

**Submission ID:** 254878165

**File name:** 57479\_Biniyam\_Memberu\_Lego\_Spike\_Int.\_Pro.\_660190\_1875536906.docx (719.94K)

**Word count:** 2321

**Character count:** 14088



**Prifysgol Cymru**  
Y Drindod Dewi Sant  
**University of Wales**  
Trinity Saint David

**Module: Integrated Project (ACCA4008)**

**Programme: Foundation (Computing)**

**Assignment title: Lego Spike Prime Project**

**Lecture: Divya Chidambaram**

**Biniyam Memberu**

**Student Name:**

**ID Number: 2405357**

## Contents

|                                  |    |
|----------------------------------|----|
| 1. Introduction .....            | 2  |
| 2. Objectives .....              | 2  |
| 3. Background Research.....      | 3  |
| 3.1 LEGO Spike Prime .....       | 3  |
| 4. Project Plan.....             | 4  |
| 5. Design Process .....          | 5  |
| 5.1 Robot Design .....           | 5  |
| 5.2 Code Design .....            | 6  |
| 5.2.1 Line following .....       | 7  |
| 5.2.2 Maze navigation .....      | 8  |
| 5.3 Results and Evaluation ..... | 10 |
| 5.4 Reflection.....              | 10 |
| 6. Conclusion .....              | 11 |
| References.....                  | 12 |

# 1. Introduction

Robotics has become a transformative field that combines engineering, computer science, and design to develop machines capable of executing tasks independently. The combination of programming techniques and construction methodologies has led to further advancements in robotics, enabling innovative solutions to complex challenges. Robotics education is a vital component of contemporary learning, as it teaches students problem-solving skills, logical thinking, and hands-on engineering experience. This platform serves as an effective method for students to explore fundamental principles in physics, mathematics, computer science, and mechanical design. This helps as a bridge between multidisciplinary courses. The projects' collaboration demonstrates how experts from different fields work together to address real-world problems.

This report focuses on the tools, techniques,  methodologies used in robotics projects, with a specific emphasis on LEGO Spike Prime, a versatile educational platform for teaching robotics and programming concepts. To this end, this paper examines hardware components, software tools, and construction techniques to provide a comprehensive understanding of the fundamental elements that underpin robotics innovation. The project explores the basic principles of robotic maze solving through the design and programming of a LEGO SPIKE Prime robot that can autonomously navigate a predetermined path. The robot must accomplish three main tasks: the first task is to follow a taped line with accuracy, the second task is to detect and turn around obstacles, and the last task is to solve a 3D maze to reach a specific endpoint.

The project begins with an in-depth analysis of the primary techniques, including line following, obstacle avoidance, and maze solving, which form the core of the project. The following sections outline the project plan, which establishes a clear structure for managing timelines, assessing risks, and implementing phased development. The design process provides more detail about the hardware assembly and software logic, which is explained through a word block to help clarify the underlying concepts. The evaluation and results section includes performance metrics, challenges faced, and potential areas for improvement. The conclusion summarizes the main findings of the project and suggests directions for future work.

## 2. Objectives

The primary goals of this project are:

1. To design and implement a suitable robot for the project.
2. To develop a reliable line-following
3. To implement obstacle detection and avoidance
4. To design an efficient maze-solving strategy.

### 3. Background Research

Robotics research encompasses multiple fields, including motion planning, human-robot interaction, and autonomous navigation. The Studies in Robotics Journal demonstrates that mechanical design and identification play crucial roles in enhancing the dynamics of robotic applications[1].

The development of autonomous robotics has undergone a substantial transformation due to advancements in sensor technology and improvements in control algorithms. The objectives of this project align with the importance of feedback control systems, mainly when applied to maze-solving applications [2].

Robotics relies on programming to enable machines to perform tasks with high precision and accuracy. Innovative programming methodologies, software development, and computational thinking-based problem-solving. The development of efficient and adaptable robotic systems requires these techniques [3].

The construction of robotic systems demands careful attention to detail, combining mechanical engineering principles with creative design strategies that offer valuable insights into construction methodologies adaptable for robotics projects [4].

#### 3.1 LEGO Spike Prime

The LEGO Spike Prime platform offers a user-friendly environment for students to learn robotics concepts and programming fundamentals. The platform enables users to build with colourful elements using drag-and-drop Scratch-based coding and provides advanced hardware sensors and motors. The platform's modular design and curriculum-aligned lessons make it ideal for exploring STEAM concepts and developing problem-solving skills. The Prime resource provides step-by-step instructions for building and programming LEGO Spike Prime robots. The SPIKE Prime Programming Guide explains threshold-based programming through sensor applications and control coding approaches [5].

The *LEGO SPIKE Prime Hardware Manual* details sensor specifications:



- **Colour Sensor:** Reflectivity range of 0–100%, used for line detection [5].
- **Ultrasonic Sensor:** 3–250 cm range, used for obstacle detection [5].
- **Gyro Sensor:** Measures angular velocity, enhancing turn precision [5].

The *LEGO SPIKE Prime software* details specifications:

- **Block Coding:** LEGO's drag-and-drop interface simplifies PID implementation [5].
- **Python Mode:** Offers advanced functionality but was not utilized due to scope constraints [5].

## 4. Project Plan

A phased approach, combined with a structured timeline, was implemented to ensure the successful completion of the LEGO SPIKE Prime robotics project. The project timeline spans ten weeks, with each phase focusing on distinct project elements while enabling multiple testing cycles, integration steps, and review periods.

- **Introduction:** The project begins with an introduction to the LEGO SPIKE Prime robotics platform and its application in developing essential problem-solving and programming skills. The project's objectives are defined, emphasizing the importance of building a robust and adaptable system.

- **Background Research.** The selection of appropriate hardware components, such as sensors, motors, and microcontrollers, is researched to ensure compatibility with the LEGO SPIKE Prime system. The evaluation of suitable software tools and techniques for creating an effective solution is also conducted.

- **Designing and Planning:** The robot construction begins with the implementation of the chosen components and tools. Each component undergoes comprehensive testing to verify its operational status. A thorough plan is created to synchronize hardware and software design with project objectives.

- **Line Following.** The line-following functionality is implemented as the first step. Integration testing is performed to ensure proper communication between hardware and software. The PID controller has been developed and calibrated, allowing the robot to follow curves with precision.

- **Obstacle Avoidance:** The robot's ability to detect and avoid obstacles is introduced. The ultrasonic sensor is integrated, and detour logic is tested to ensure the robot can navigate around barriers effectively.

- **Maze Navigation:** The robot's maze-solving capabilities are developed and tested within a 3D maze environment. The robot utilizes algorithms to navigate and autonomously reach its destination.

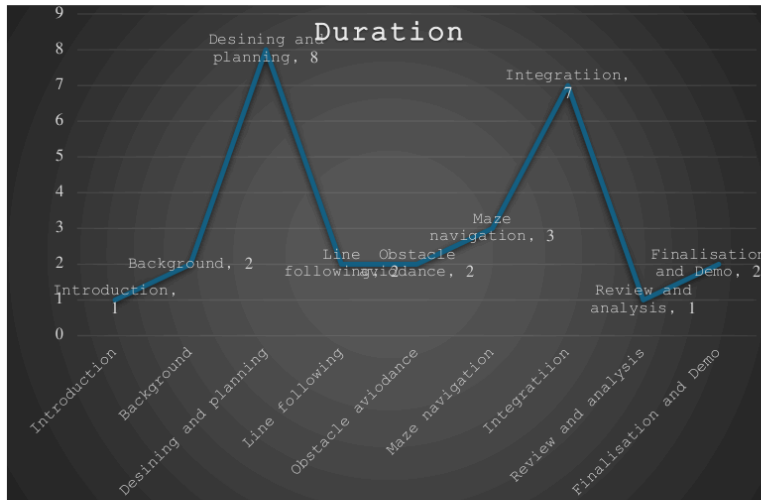
- **Integration.** All stages of the project are integrated to form a cohesive system. All functionalities operate smoothly through debugging, which resolves any inconsistencies.

- **Review and Analysis.** The project is reviewed in its entirety. Results are documented, and the robot's performance is analysed. The development process challenges are evaluated, and potential improvements are identified.

- **Demonstration and Finalization:** The final design of the robot is complete. The operational guidelines and support documentation for users are prepared. A presentation is created to showcase the project's features, outcomes, and key achievements.

| Project Plan           | Start Date | End Date   | Duration |
|------------------------|------------|------------|----------|
| Introduction           | 29/01/2025 | 30/01/2025 | 1        |
| Background             | 05/02/2025 | 07/02/2025 | 2        |
| Designing and planning | 12/02/2025 | 21/02/2025 | 8        |
| Line following         | 26/02/2025 | 28/02/2025 | 2        |
| Obstacle avoidance     | 05/03/2025 | 07/03/2025 | 2        |
| Maze navigation        | 12/03/2025 | 15/03/2025 | 3        |
| Integration            | 19/03/2025 | 26/03/2025 | 7        |
| Review and analysis    | 02/04/2025 | 03/04/2025 | 1        |
| Finalisation and Demo  | 03/04/2025 | 05/04/2025 | 2        |

Table for Project Plan



Gantt Chart for the Project Plan

3

## 5. Design Process

This project is divided into two key stages. It is divided into hardware and software design. The hardware design involves constructing the robot, while the software consists of coding the program.

### 5.1 Robot Design

The first step is to make a suitable robot for the project. The robot is designed to be stable, move smoothly, and has all the necessary sensors to function correctly. The design of the robot is like that of the drive base. We encountered difficulties in determining the most suitable design for the project. We prefer to choose a drive base robot to navigate a small area

with narrow, complex, and obstacle-filled paths. It is equipped with colour, ultrasonic, and gyro sensors to help it navigate around the 3D maze.



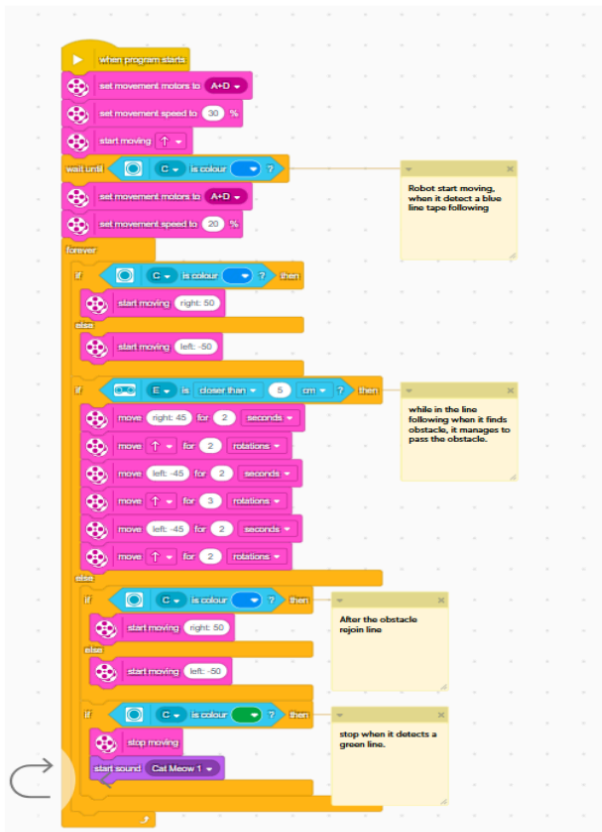
*Drive base robot design for the project.*

## 5.2 Code Design

There are two types of coding suited to create and employ Lego robots. Those are block-based and text-based programming languages, such as Python. Block-based programming is preferable for us, as it is a visual language that uses drag-and-drop blocks to write code and is suitable for designing programs that involve line following and maze navigation.

### 5.2.1 Line following

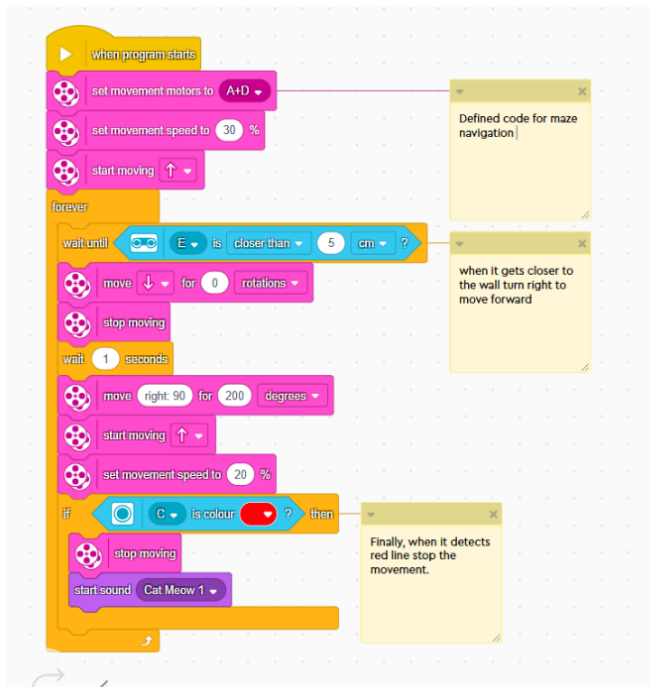
The initial action for maze navigation requires the robot to follow the line. The robot needs to track the taped path while detecting the block obstacle and then resume following the path. We implement Word Block as a code-based programming tool to analyse colour sensor data, which enables line-following logic through loop operations. The system uses ultrasonic sensors to measure distances for obstacle avoidance while executing programmed movements. The robots move at 20% speed to track the tape precisely through colour sensor operations until the distance sensor detects objects within a 5 cm range. Then, the robot stops and turns right to pass the obstruction before rejoining the tape path using colour sensor data. The colour sensor stops the robot through its programming after detecting the green colour, while producing a sound. The entire process, including programming, is illustrated in the following diagram.



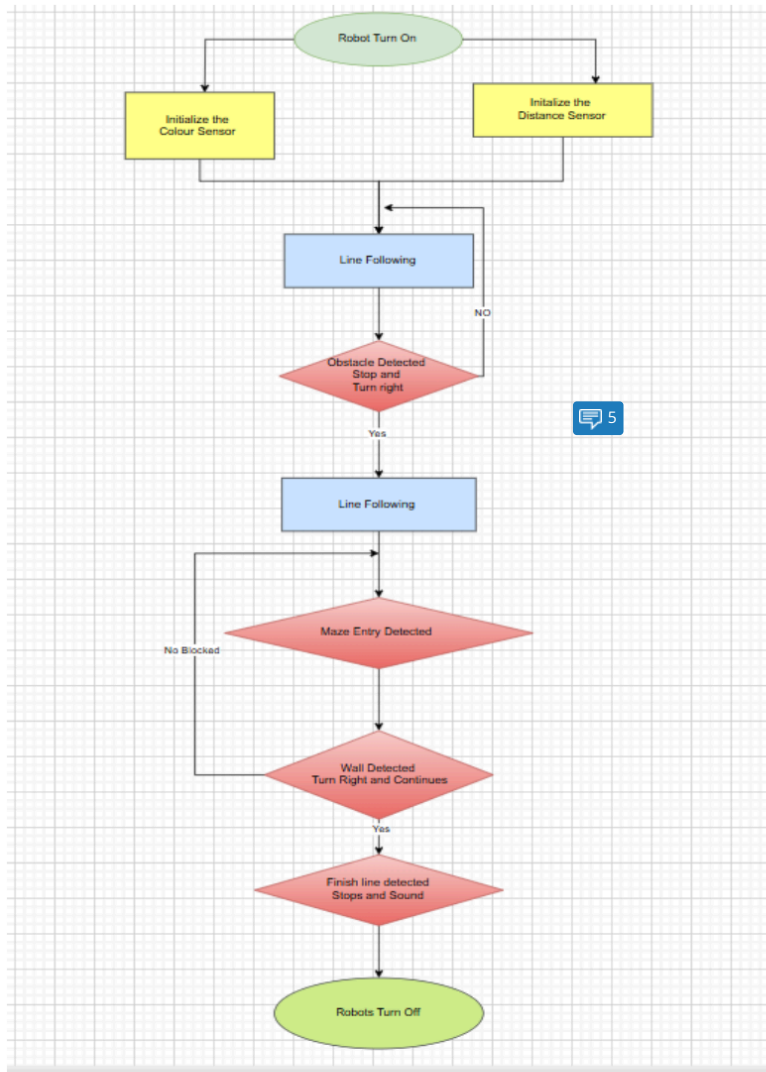
Line following block of code

### 5.2.2 Maze navigation

The Maze comes after the completion of the line. The robot must navigate the maze independently until it reaches the finish point, playing a tune to signal its success. We use Word Block as our programming tool to connect distance sensors, which trigger stop, steer, pause, turn, and loop resumption actions. We utilize ultrasonic sensors through the right-hand rule. The stop-and-steer approach was implemented because it provides reliable wall detection. The block code diagram indicates that the robot's movement is set to 20% forward, which corresponds to its idle speed. The robots pause and turn right whenever they detect a wall within 5 cm of distance in front of them before continuing their movement toward the junction. The block of code operates in an infinite loop, which causes the robot to perform a 90-degree right turn. The robot turns around when it detects a dead end and then continues moving forward until it reaches the finishing point, using the colour sensor, this triggers a sound through the SPIKE Prime speaker when it encounters the red block.



Maze navigation block of code



*Flowchart for the integrated project*

## 5.3 Results and Evaluation



The robot was successfully built and achieved after many trials. Building a real-world structural reliability robot can be time-consuming. A detailed hardware specification for constructing a standardised robot was assembled correctly, and all electrical connections functioned adequately, including those for sensory components. The robot design is suitable for maze navigation, a drive base robot is simple to navigate in a small area with narrow, complex, and obstacle-filled paths. The physical inspection during the maze run was performed well, aligning with the project's goal; it was built and executed effectively.

The primary goal of the project is to identify the most effective code for the program. We utilize a block-based programming language, a visual language that employs drag-and-drop blocks to facilitate code writing. In the line that follows, it sometimes fails to require the line again because of turning around the obstacle. The line-following accuracy was successfully achieved, but needs some corrections compared to the project goal.

The block-based programming function works well for maze navigation, but the right-hand rule fails to turn in the dead zone and needs smooth correction, though it is reliable for precise turns in a broader area. When using block-based approaches, it is essential to be more systematic in debugging complex systems; otherwise, the project's accuracy may be compromised.

The separation of code into distinct sections for each problem enables us to identify the issue quickly and reduces debugging time, allowing us to implement standardised tests more frequently.

## 5.4 Reflection

The project likely provided valuable insights into robotics, programming, and construction methods. Working with LEGO Spike Prime likely helped you develop problem-solving skills while enhancing your understanding of hardware-software integration and your ability to overcome challenges.

The assembly of hardware components proved to be challenging because it required both accuracy and patience. Debugging and troubleshooting software errors. The robot's performance needed to be consistent throughout testing. The obstacles we faced would have tested your patience, as well as your adaptability and analytical thinking.

The project implemented several aspects that probably contributed to efficient teamwork and collaboration. The project followed a precise plan and execution, including a problem-solving approach and a well-defined timeline. The robotic system performed reliably because hardware and software components worked well together.

Future projects require additional time allocation for iterative testing and refinement to handle unexpected issues. Research new tools or techniques which can help optimize hardware assembly and programming workflows. Document lessons learned to apply them to subsequent robotics projects.

## 6. Conclusion



This project transformed theoretical robotics concepts into a tangible experience. While the final robot met all core requirements, the journey revealed several important insights. The project highlighted the importance of strategic planning, interdisciplinary knowledge, and iterative problem-solving. Key achievements included in-line tracking accuracy and maze success. Challenges in navigation precision emphasized the value of resilience and teamwork in overcoming them. For future projects, a more in-depth exploration of advanced programming techniques and expanded testing scenarios will further enhance performance, providing an excellent foundation for advanced robotics work.

## References

- [1] P. Righettini, "Real-Time MEMS-Based Joint Friction Identification for Enhanced Dynamic Performance in Robotic Applications," MDPI Robotics, 21 March 2025. [Online]. Available: <https://www.mdpi.com/>. [Accessed 29 03 2025].
- [2] Y. Xie, "Advanced Sensing and Control Technologies for Autonomous Robots," MDPI Sensor, 23 Aug 2024. [Online]. Available: <https://www.mdpi.com/>. [Accessed 29 03 2025].
- [3] A. Owen-Hill, "What is the Best Programming Language for Robotics?," Robotiq, 19 Dec 2024. [Online]. Available: <https://blog.robotiq.com/what-is-the-best-programming-language-for-robotics>. [Accessed 19 03 2025].
- [4] C. Woods, "The Future of Construction Robots in 2025: Automation, Innovation, and Efficiency," Boom and Bucket, 02 04 2025. [Online]. Available: <https://www.boomandbucket.com/blog/construction-robots>. [Accessed 03 04 2025].
- [5] "Lego Education Exploring Spike," Lego, [Online]. Available: <https://community.legoeducation.com/>. [Accessed 25 03 2025].

75/100

PAGE 1

PAGE 2

PAGE 3



**Comment 1** | Transferable Skills(15)

challenge effectively explained and aligns with broader learning objectives

PAGE 4



**Comment 2** | Engagement with Literature(15)

application of research to design decisions could be more critically connected. some figures of lego education kit could be added

PAGE 5

PAGE 6



**Comment 3** | Transferable Skills(15)

phase-based timeline included This meets expectations well. Slight improvement could come from stating your individual leadership role in a particular phase or section.

PAGE 7

PAGE 8

PAGE 9



**Comment 4** | Knowledge and Understanding(40)

design process includes both hardware and software explanations. Sensor calibration and testing strategy could be deeply explored.

PAGE 10



**Comment 5** | Knowledge and Understanding(40)

well done

PAGE 11



**Comment 6** | Cognitive and Intellectual(30)

Good reflection on successes and problems

PAGE 12



**Comment 7** | Cognitive and Intellectual(30)

A brief discussion of limitations or possible alternatives (e.g., how sensor layout could be improved) would add depth.

PAGE 13

GRADING FORM: DC-ACHA4008

**BINIYAM MEMBERU**

**75**

**ENGAGEMENT WITH LITERATURE(15)**



**11**

**KNOWLEDGE AND UNDERSTANDING(40)**



**28**

**COGNITIVE AND INTELLECTUAL(30)**



**24**

**TRANSFERABLE SKILLS(15)**

