MicroMouse Maze Runner

DESIGN DOCUMENT

Team number: sddec20-28
Client: Dr. Phillip Jones
Advisor(s): Dr. Phillip Jones

Team Members/Roles
Joshua Wooi
Austin Chesmore
Aaron Walter
Richard Anderson
Tyler Fuchs
Jorge Gomez

Team Email
sddec20-28@iastate.edu

Team Website
sddec20-28.sd.ece.iastate.edu

Revised: 2020-02-23
Version: 1

Executive Summary

Development Standards and Practices Used

- C Programming
- Agile, Kanban Boards
- Deliverables

Summary of Requirements

- Functioning MicroMouse prototype unit(s)
- Embedded software algorithm for autonomous function
- Wireless feedback
- Reconfigurable Maze Design

Applicable Courses from Iowa State University

- EE 201
- EE 285
- CprE 288
- EE 330
- EE 333
- Com S 309
- Com S 327
- Com S 311

New Skills Acquired That Was Not Taught in Courses

- Teamwork
- Website Development
- Git
- PCB Design
- Client Meetings / Updates
- Interpersonal Skills

Table of Contents

1. Introduction

1.1 Acknowledgement

This section acknowledges Dr. Philip Jones for his contributions in providing technical advice to the team and Lee Harker for his guidance in sourcing hardware parts and using the EAGLE software.

1.2 Problem & Project Statement

Problem Statement:
Lack of interest in/general exposure to Robotics.

Solution Approach:
A MicroMouse Maze Runner Showcase.

In the near future there are projected to be many more computer/electrical/software based jobs than there are people to fill the jobs, so we need to try to give a reason for future ISU students to consider robotics based curriculums.

This project serves to showcase the capabilities of engineering students from multiple disciplines at Iowa State University.

This project describes designing a MicroMouse, a small wireless robot unit capable of traversing through a maze using hardware components controlled by software algorithms.

The expected outcome of this project is a functioning MicroMouse unit(s), capable of both autonomous and/or remote control functions, that can traverse through several different maze designs effectively.

1.3 Operational Environment

The end product of this project will operate in a controlled environment of a smooth surface in a sturdy maze, situated indoors. The operation of a MicroMouse outside of this environment condition is unexpected and undesired.

The environment will be expected to be modular and transportation friendly, so while the product cannot just be used anywhere, it can be used anywhere the micromouse maze can be transported to.

1.4 Requirements

Functional requirements of this project would be that the mouse is able to move through the maze, is able to map a maze to a maze model, is able to find the shortest path of the maze model, and is able to follow paths through the maze. We also must have a maze that can fit the micro mouse, be modular, transportable, and solvable by the mouse.

Non-Functional requirements of the project is that the mouse should be accurate, reliable, and sturdy. Our maze will have to be of good quality materials so that it is able to last long as it is transported and reconfigured.

Economic requirements for the project would be that we stay within our $500 budget, and that we chose good quality parts for what we pay for.

Environmental requirements would be that we have a table to set the maze up on, and that we have a maze for the micromouse to solve.

UI requirements would be that we want the mouse to be easy to use for the recruiters or users showing it off. This means that we will need to have clear documentation on usage, and clear button and control placement on the physical mouse.

1.5 Intended Users & Uses

As a showcase project, the intended user of a MicroMouse is anyone with latent interests in Robotics. This project is designed to inspire and encourage anyone to pursue this further and get involved in the STEM field.

1.6 Assumptions & Limitations

Assumptions:
This micromouse will be used to demonstrate how an autonomous MicroMouse robot runs through a maze. It will not be used for competition. The MicroMouse will be half sized and can not exceed the palm of one's hand.

Limitations:
A strict budget of $500 has been set by the client. Mouse must be able to navigate the maze without user input.

1.7 Expected End Product & Deliverables

Product ordering (2/21):
Initial parts order to be reviewed and approved by ETG so that progress can continually be made for Prototype 1.

Prototype 1 (3/31):

At the end of March we have a first working prototype due for our client. This prototype should be able to move and run on its own while detecting if it is going to hit a wall or not.

2. Specifications & Analysis

2.1 Proposed Approach

Team decided to protype the project using an Adafruit Feather as a microprocessor, to allow for less possibility of project bugs and simpler implementation.

In case of potential errors or hardware malfunctions occurring while building working models, ordering spare components will ensure the project progress won't be delayed

2.2 Design Analysis

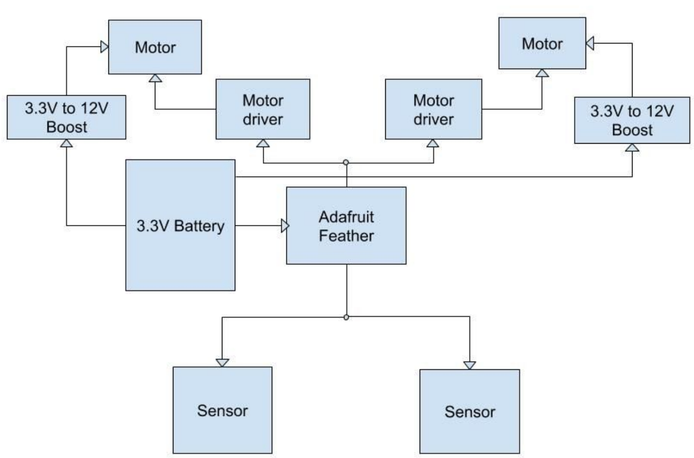
The team has thoroughly researched the specifications that the Micromouse project will need to meet. Through this research, a project components list was created. These different components then be assessed by Dr. Jones and ETG to ensure that nothing is done needlessly.

The methodology the team is employing is currently very effective and should allow for an effective transition into from the prototyping phase into the final design construction.

2.3 Development Process

Developing this robot will initially use a Kanban style where there is a list of problems with different priorities, then as the mouse reaches completion we will move to a waterfall style where documentation will be a main priority. This will help the team in the future with the next version of prototypes.

2.4 Conceptual Sketch



3. Statement of Work

3.1 Previous Work & Literature

Classwork here at Iowa State has laid a foundation that we can help build our project upon, specifically 288 in it's design of sensors on an autonomous robot. Since this project has been done many times before, there are plenty of online articles and documentation that will help us in the designing and creation of the micromouse. For our pcb design we utilized a software program called eagle and used this tutorial to learn about the program.

3.2 Technology Considerations

The technology that will be incorporated into the micromouse will be cost effective and modulor. Each component was selected to be easily integrated into the prototype design. By using off the shelf components the project will have less likely to have quality issues, such as solder bridging or improper design. This removes several potential failure points, allowing for the project to be bug tested more effectively. However, one weakness of using these types of components is that there is less controllability of the specification.

3.3 Task Decomposition

Different programs will be essential to a successful project Step one will include learning about the Eagle software to develop our own PCB board. The following steps are needed to finish the first prototype; ordering parts, configuring parts to work in tandem, building software to use sensors successfully, and finally putting them together in a shell to have the mouse be able to move on it own.

3.4 Possible Risks & Risk Management

Prototyping the mouse will be the most worrisome of our project, getting all the different pieces to work together included. After we learn what hardware can work with what software prototyping will become much easier and we should be able to push new designs faster. The cost of the maze can range from $10s so $100s of dollars so keeping an eye on pricing is another concern. Our last concern is PCB design, since no one on the team has knowledge in PCB design or the Eagle software that is an area that someone will have to learn.

3.5 Project Proposed Milestones & Evaluation Criteria

Key milestones are mainly progress checkpoint. Checkpoint one will be a mouse that is able to move. Followed by the mouse being able to detect a wall, then the mouse being able to navigate the maze. Finally, the mouse is able to navigate the maze and choose the fastest path.

3.6 Project Tracking Procedures

We have a shared calendar that already has deadlines marked and our advisor can see it to make sure that we are progressing at a fair rate. Each member is responsible for a different part of the development process and therefore will be held accountable by his teams for their contribution.

3.7 Expected Results & Validation

The desired outcome as outlined by the client is a half size micro mouse that can run a maze on its own. It will also have a display that will be used to run the program and have the ability to respond to user input. The user will have the ability to control the mouse with a controller and the mouse will adequately respond. These results will be easy to validate as they are just tested by a user when the product is complete.

4. Timeline, Resources, & Challenges

4.1 Project Timeline

Range of Time	Achievement
1/20/20 to 1/26/20	• Set up a system of communication between team members • Set up initial meeting time and place to discuss projet • Set up a meeting with our client. • Met with client to discuss project expectations.
1/27/20 to 2/02/20	• Discussed what we would need for the project parts • started research on project
2/03/20 to 2/09/20	• Meeting with client was rescheduled
2/10/20 to 2/16/20	• Started to create a parts lists for the protype
2/17/20 to 2/23/20	• Second meeting with a client • Reviewing parts list to finalize parts.

4.2 Feasibility Assessment

Final Requirements Concerns

- Autonomous maze navigation
    - Wall detection without getting stuck in a loop
    - Detecting the goal of the maze
- User input controls
    - Bluetooth protection from other devices
- Maze Design
    - Price - keeping the price in check as budget can get blown away by the maze
    - Mobility - being able to transport the maze while also being large enough to be complex

4.3 Personnel Effort Requirements

Member	Tasks	Hours	Concerns
Aaron	Maze running algorithm	6/hours a week	Sensors
Austin	PCB	5/hours a week	Making PCB
Jorge	Sensors	5/hours a week	ToF Sensor testing
Joshua	Hardware sensors	5/hours a week	learning hardware sensors
Richard	Maze running algorithm	5/hours a week	Functionality, Readability
Tyler	Maze	5/hours a week	Mobility

4.4 Other Resource Requirements

The main outside concern is where we can store the maze while it is being built. As per the client, they would like a maze that is at least 9 feet by 9 feet, so this will require a large open space.

4.5 Financial Requirements

$1000,0000.00 big boi budget

5. Testing and Implementation

5.1 Testing Plan

5.1.1 Needed Tests

Unit testing would include testing the shortest path search algorithm. If we have multiple helper functions included in the path search algorithm then we could individually test that those functions return expected values.

We will also likely have multiple functional tests testing that the algorithm truly returns the shortest path of different mazes.

5.1.2 Items to be Tested

Will need to test hardware components and see if results are expected and consistent, eg the feather, moters, encoders, do what they are expected to do.

Need to test software, specifically that pathfinding and maze mapping return expected results.

5.1.3 Test Cases

TestMazeCorrectPath() : input is null, testing procedure is mock a maze, run it through the solve functionality, and assert the result is what should be expected.

TestMazeMapping() : input is null, testing procedure mocks mouse turning and movements, which should be done with a function that maps maze based on movement.

Must assert that the result of the mapping function is what's expected.

5.1.4 Anticipated Results

Anticipated test results for unmodified, previously successful functionality will be successful, while modified functionality will need to be tested again, and might anticipate success, but could be failure.

5.1.5 Performed Tests

TODO when were able to make tests

5.1.6 Evaluation

TODO when were able to make tests

5.1.7 Changes Because of Testing

TODO when were able to make tests

5.1.8 Retesting

TODO when were able to make tests

5.1.9 Documentation

TODO when were able to make tests

5.2 Interface Specifications

TODO when working on testing

5.3 Hardware and software

TODO when working on testing

5.4 Functional Testing

TODO when working on testing

5.5 Non-Functional Testing

TODO when working on testing

5.6 Process

TODO when working on testing

5.7 Results

TODO No test results yet

6. Closing Material

6.1 Conclusion

The work we have done so far includes research for each part of the mouse and where we can order it from.

6.2 References

Blom, Jim. "Using Eagle: Schematic." Using EAGLE: Schematic, learn.sparkfun.com/tutorials/using-eagle-schematic/all.

6.3 Appendices