



## **Micro-Mouse Design Project**

### **Engineering Team:**

Sami Elias  
Evan Jina  
Willy Herrera  
Jeff McGarrah

Sponsored by: Dr Gordon Lee

**Submitted to:** John Kennedy and Lal Tummala  
**Design Co. Ltd, San Diego, CA**



## Table of Contents:

1. Abstract: .....	3
2. Problem Definition: .....	3
2.1 Functionality: .....	3
2.2 Customer Needs: .....	5
3. System Design: .....	6
3.1 Background Research: .....	6
3.2 Software: .....	7
3.3 Constraints: .....	8
3.4 Illustrations: .....	8
4. Testing and Verifications: .....	9
4.1 Benchmarks: .....	10
5. Project Management: .....	10
5.1 Project Plan: .....	10
5.2 Cost Analysis: .....	12
6. References: .....	13
7. Promotional Flyer: .....	14

## **1 Abstract:**

Micromouse is an event where small mechanical robots compete in finding the center of a maze. The robots must be completely autonomous which means that no external interaction with the robot is allowed. Besides the robot itself, there are general rules for the maze itself. Each maze consists of 16x16 cells. Each cell is 18x18 cm<sup>2</sup> and the height of the wall is 5cm. Generally the robots are equipped with sensors for navigation and wall sensing as well as a clever algorithm. Modifications of Bellman's Floodfill algorithm are widely used in Micromouse competitions. The algorithm as the name suggests stores all cells that are visited. This establishes a map of the maze and therefore once the center point is reached, the shortest route to the starting position can be easily retrieved. It is a very open competition which allows many modifications to both hardware and software yet it remains challenging to achieve an efficient system which can satisfy all constraints. The Micromouse competition was adopted by several countries including the U.S in the late 1970's.

## **2 Problem Definition:**

Design an autonomous ground based robot which will be able to find the center of the maze where it must retrieve an infrared signal and display it on an internal LCD screen. Thereafter the robot must return to its starting position. All of the above should be completed at the shortest possible time. The robots intelligence must be self-sustained and physically attached to the robot.

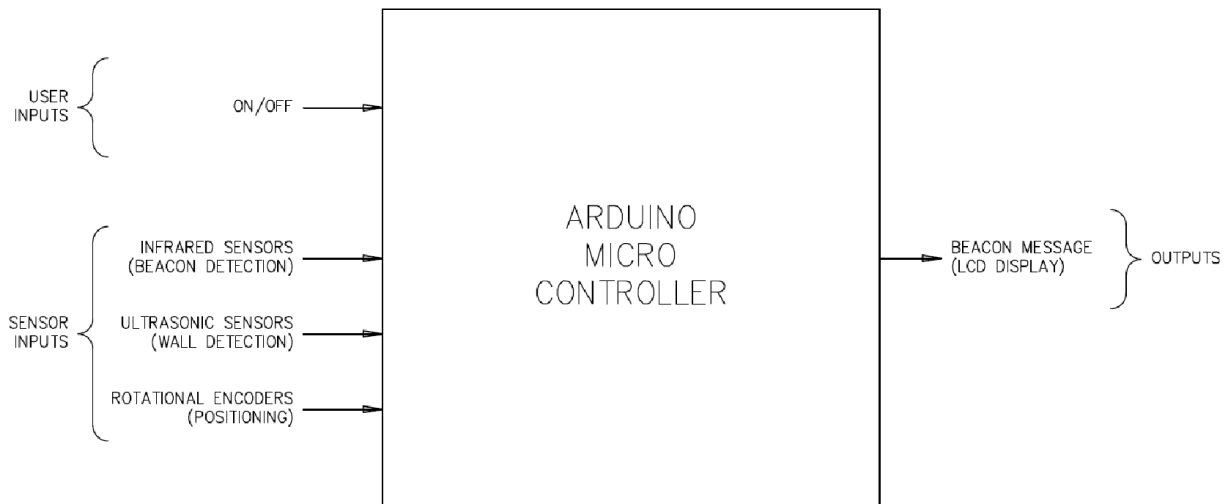
### **2.1 Functionality:**

There are several functionalities that the robot must possess in order to achieve its goal. Wall detection and minor center point adjustments will be handled by three Ultrasonic sensors. The orientations that will be covered are leftward, rightward, and forward. At any time the ultra-sonic sensors must be able to provide the microcontroller

with discrete distances to the walls from which the micro controller will either adjust its center point or determine direction. In order for the mouse to record and travel accurately cell by cell, it needs some kind of motor feedback system for the micro controller to know how many pulses or how many rotations it should expect from the motors in order to reach a single cell. For this, rotational encoders will be attached to the DC motors in order to register the amount of revolutions that are required to do the following:

1. Position the robot from one cell to the next
2. Allow the robot to make an accurate 90 degree turn or a 180 degree turn in a dead end situation

Together with the ultrasonic sensors, this establishes the foundation of robot positioning within the maze. The microcontroller should have enough memory to keep track of all the visited cells in order to establish a map. A rough estimate of the memory requirement is about 4k SRAM. For this project an Arduino Due is used which consists of an ARM Cortex processor with 96k of SRAM. Once the center of the maze is reached the robot can pick the shortest path found from its memory back to the starting location. Before returning to the starting position, it is required for the robot to decode the IR signal that is emitted by a beacon which is positioned at the center point of the maze. The decoded signal has to be displayed on an LCD which allows the observer to see the message once the robot has returned to its starting position.



*Figure 1: Functionality Diagram*

## 2.2 Customer Needs Q&A:

- What is the problem to be solved?

Our task is to assemble a small, autonomous robot which can navigate an unknown maze in the fastest time possible. Upon reaching the end of the maze, the robot must detect an infrared signal, then return to the start of the maze and display the signal visually to be interpreted by spectators.
- Why is there a problem?

Since the robot is fully autonomous, sensors must be implemented to guide the robot in a sensible direction in order to properly navigate the maze. With proximity sensing, the robot should then be able to understand when turns are available to take, but it must use an algorithmic approach to calculate when and where it must go. Furthermore, it must store its current path else face unintended backtracking, so memory can be an issue.
- How will I know when I am done?

The end result is when the robot returns to the start of the maze with the correct signal from the finish displayed for visual interpretation, such as on an LCD screen. Until the robot can accomplish this, the task is unfinished.
- What is the upper limit cost to do the design?

The cost is anticipated not to exceed \$250.
- What are the consequences of the system failing once in operation?

The only consequence for failure in operation is a loss of one trial in a competition due to the timing constraint. Once the trial ends, successive trials can proceed after maintenance.
- How will it be determined that the design is acceptable?

So long as the design does not infringe the rules of the competition as outlined by the official IEEE regulations, it will be deemed acceptable for use.

### 3 System Design

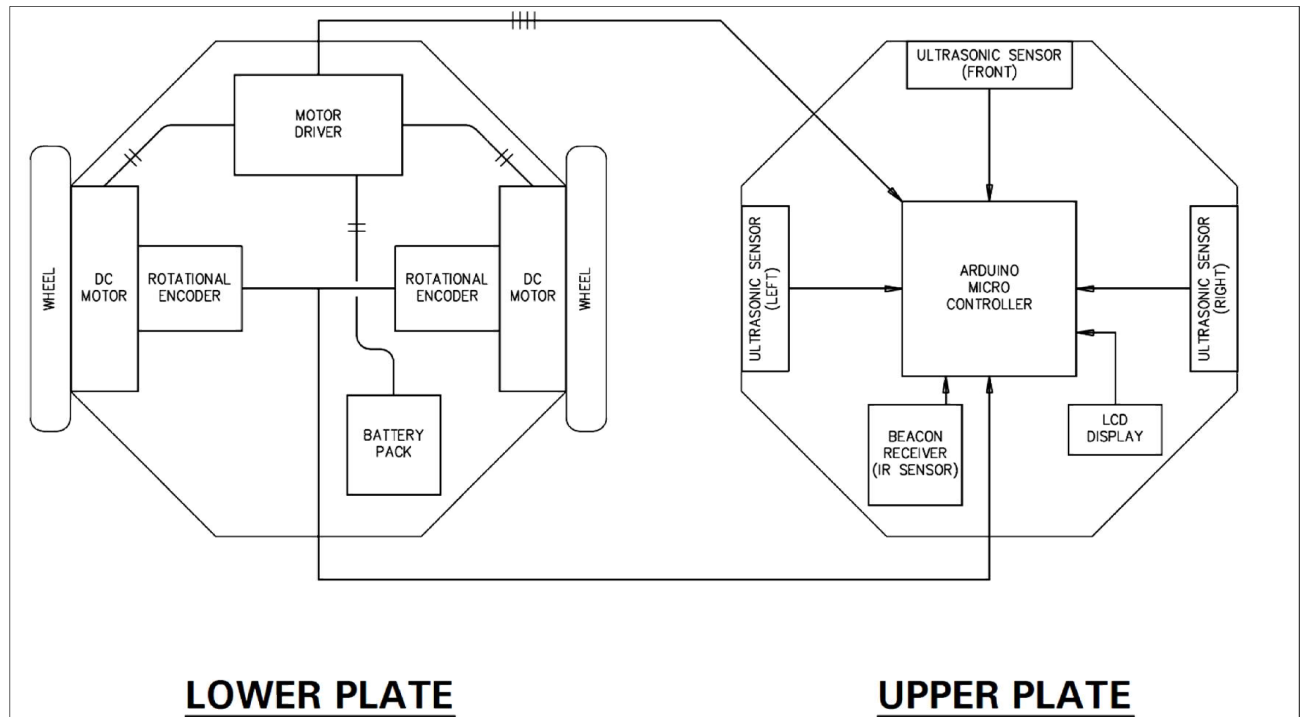


Figure 2: System Design Block Diagram

#### 3.1 Background Research

There have been many previous designs of different Micromouse robots since the competitions started in the late 1970's. Designers have many options and technologies to choose from as long as the constraints are met. When it comes to wall detection, the most common used sensor is the infrared sensor from Sharp inc. These sensors have unit costs of about \$15 dollars and are very easy to use. They come in two versions, a long and a short range version. The long range version would qualify as the front end sensor where it can detect walls at ranges from 10cm to 80cm. The downside with this technology is that it's sensitive to ambient light interference. Any dark spots would also absorb the IR lighting and therefore prevent the IR receiver from getting any usable data. A different technological approach would be to use ultrasonic distance sensors. Many designs including ours use ultrasonic waves to calculate the distance the sound wave has traveled. The benefit of using ultrasonic sensors is that

they are accurate even within 2cm. Furthermore the technology is not sensitive to any ambient lightning or dark spots. The noticeable downside is not directly related to any technological mean but to the unit cost.

Apart from wall detection, the Micromouse has to be able to navigate deterministically throughout the maze. This means that many designs have always had some kind of a feedback system incorporated to their motor system. There are two major well known systems that can provide feedback about positioning. The stepper motor has fixed amount of steps which allows it to rotate in correspondence to the amount of pulses it receives from the microcontroller. This technology is simple and easy to use. However the downside is that these types of motors are usually heavy, expensive and slow. On the other hand there are the dc motors which by themselves cannot give any useful feedback to the microcontroller, but if they have rotational encoders attached to them then the number of rotations that are registered by the encoders can be used to position the robot accurately. The main downside of using rotational encoders instead of stepper motors is the added complexity.

Finally the traditional software has consisted of variations of bell-mans flood fill algorithm which suggests that every cell that is visited to be stored in memory. Once the center point has been found, then the robot will be able to determine the shortest path back to its initial starting position. This requires that the microcontroller has enough SRAM to map the entire maze.

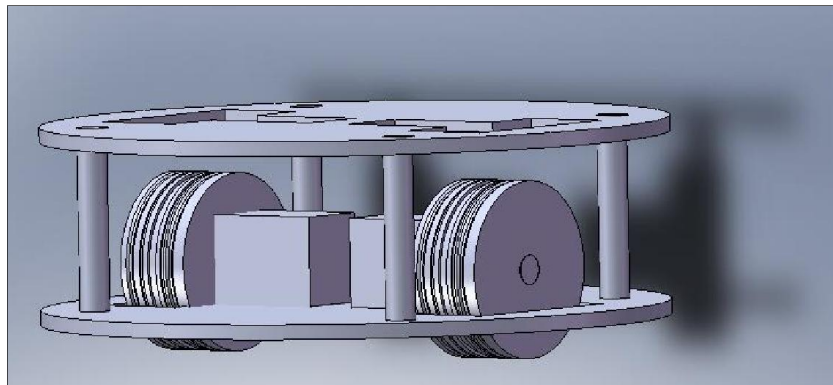
## **3.2 Software**

In order to implement our design, we must setup essential software components in order to test and program our product. Since we will be using an Arduino microcontroller, we will need to develop our code in the Arduino IDE, which supports all Arduino boards. However, due to the lack of debugging tools in the Arduino IDE, we plan to also develop in Microsoft's Visual Studio environment. By implementing an Arduino programming plugin to Visual Studio, we will be able to code using the operational procedures of the Arduino IDE and take full advantage of the Visual Studio debugger tool.

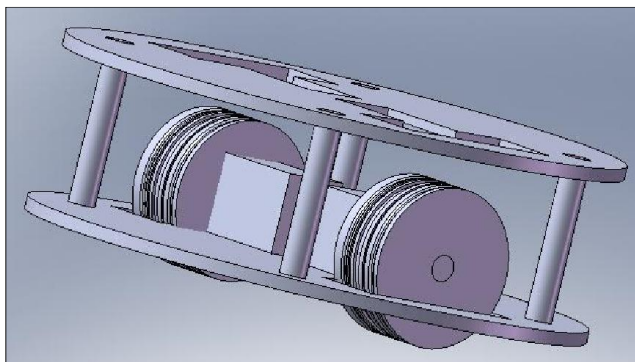
### 3.3 Constraints

There are multiple constraints that our team must take into consideration when designing and building the Micromouse, some of which are Economical and environmental constraints. Economically, our team has been afforded approximately 250 dollars, well below the 500 dollar standard set by IEEE for competitions of this type, however, we have been provided with design tools such as oscilloscopes, computers, fabrication equipment, etc. The size of the Micromouse is also constrained by the environment it must navigate, a maze measuring of approximately eight square feet and made up of cell arrays approximately six inches apart. Other constraints include manufacturability, considering the size of the mouse and the lack of funding, considerable care must be taken into account not to damage any components during manufacturing.

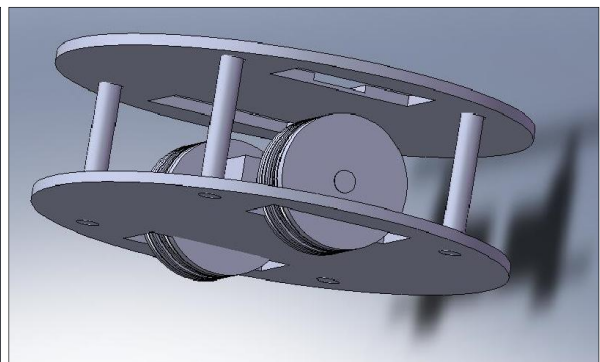
### 3.4 Illustrations



*Figure 3: Level view of robot*



*Figure 4: Right tilted view*



*Figure 5: Rear tilted view*



## 4 Testing and Verifications

The software element is crucial to our testing procedures. Each component will need to be tested to verify its functionality. After component testing, each part will be integrated into the complete design, which will lead to a series of post-integration testing. Testing after integration is expected to be an ongoing process and will serve to verify proper interfacing between components. For component testing, we expect to run the following test procedures:

- Ultrasonic Sensor Proximity Testing:

The ultrasonic sensors which serve as our wall detecting sensors need to be tested for both distance and accuracy measurements. We expect to find the limitations to these sensors, as well as compile a list of distance versus voltage measurements.

- DC Motor Benchmarking:

The DC motors will be benchmarked for speed using a function generator with a swept signal. We intend to find an optimum rate appropriate to our course.

- Rotational Encoder Calibration:

The rotational encoders must be tested for precision in order to make precise cell-by-cell movement and right-angle turns. This will be done through program interfacing with the Arduino IDE.

- LCD Display Visual Test:

The LCD display must be tested for usability. Using the Arduino IDE and microcontroller, we plan to test a simple “Hello World” program to be output on the LCD screen.

- IR Sensor:

The IR sensor for beacon detection, like the ultrasonic sensors, will need to be functionally tested to determine range and accuracy. We will compile distance versus voltage measurements.

## 4.1 Benchmarks

Action items	Progress	Deadline
❖ Description of project proposed by professor. Create team for specific project.	[Complete]	2/7/2014
❖ Research parts and items necessary to begin the design. i.e. types of sensors and motors to use, algorithm, previous designs of the Micromouse. Submit basic schematic	[Complete]	2/21/2014
❖ Propose the design our group will use. Receive all the parts ordered online. Complete proposal discussed in class outlining the system design, budget, and project plan.	[Complete]	3/6/2014
❖ Preliminary Design Review (Provide power point presentation).	[Incomplete]	3/14/2014
❖ Test each individual part and make sure everything works before incorporating into Micromouse.	[Incomplete]	3/14/2014
❖ Complete the integration of the parts to form the physical Micromouse product.	[Incomplete]	3/28/2014
❖ Critical Design Review (Provide power point presentation).	[Incomplete]	4/11/2014
❖ Test our algorithm and perform trial runs on the beacon receiver to understand necessary changes needed to make.	[Incomplete]	4/18/2014
❖ Complete Micromouse with integration of algorithm and working parts. Have plenty of trail runs on different types of mazes.	[Incomplete]	4/25/2014
❖ Micromouse Competition!	[Incomplete]	5/8/2014

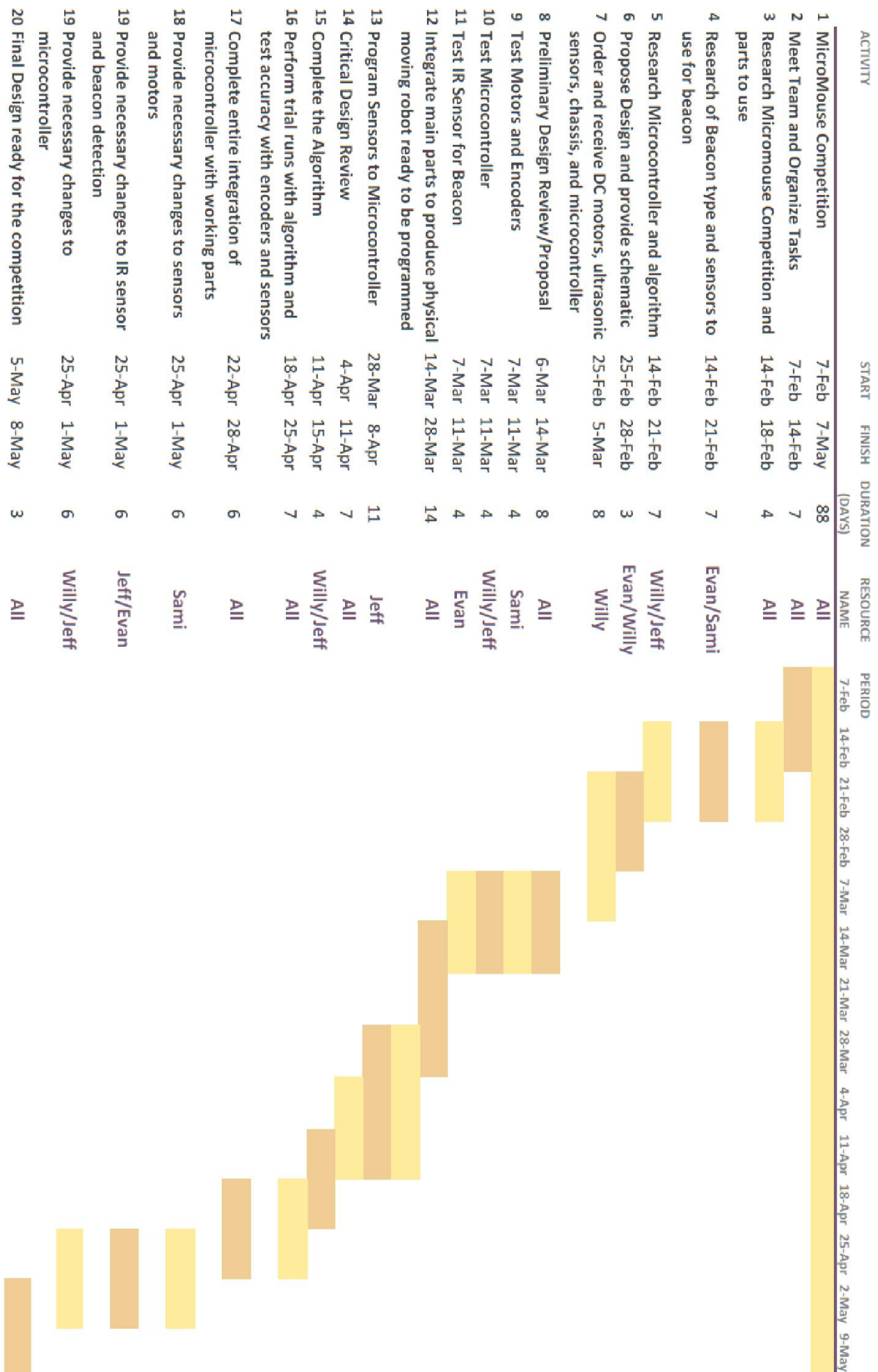
## 5 Project Management

Essential to our success is project management and planning. Our team of engineers, has spent adequate time drafting a plan that includes a Gantt chart, and cost analysis using tools such as Microsoft Excel and Microsoft Project.

### 5.1 Project Plan

The following page below includes the Gantt chart detailing our project plan

### DeadMous Gantt Chart



## 5.2 Cost Analysis

As mentioned earlier in this report, we plan the budget so as not to exceed \$250. For each part outlined in our functional diagram, we have allocated specific pricing measurements to stand within our budget. The following table outlines our proposed budget:

Part	Price Per Unit	Quantity	Shipping Fees	Subtotal
<b>127mm Chassis Plate</b>	\$8.00	2	\$0.00	\$16.00
<b>Aluminum Standoff</b>	\$2.00	1	\$0.00	\$2.00
<b>Ball Caster w/ Metal Ball</b>	\$2.00	1	\$0.00	\$2.00
<b>Wheels w/ Rotational Encoders (2)</b>	\$40.00	1	\$0.00	\$40.00
<b>DC Motor</b>	\$16.00	2	\$0.00	\$32.00
<b>Motor Driver</b>	\$8.00	1	\$0.00	\$8.00
<b>Arduino Due Microcontroller</b>	\$50.00	1	\$0.00	\$50.00
<b>Ultrasonic Sensor</b>	\$5.00	3	\$0.00	\$15.00
<b>IR Sensor</b>	\$9.00	1	\$0.00	\$9.00
<b>8x2 Character LCD Screen</b>	\$10.00	1	\$0.00	\$10.00
<b>Polymer Lithium Ion Battery</b>	\$6.00	1	\$0.00	\$6.00
Total				\$190.00
Tax at 10%				\$19.00
<b>Grand Total</b>				<b>\$209.00</b>

Table 1: Bill of material

The expected cost of the design is at \$209. This leaves \$41 remaining to expend on backup components in the case of unexpected damages. Pending the use of these resources on secondary parts for our design, we may implement additional features to the Micromouse that are not relevant to its accomplishment of its task (i.e. “the bells and whistles”).

## 6. References

1. "A.I.R.S." *A.i.r.s.* SDSU, Spring 2012. Web. 3 Mar. 2014.  
<<http://airs.sdsu.edu/index.html>>.
2. "A Few Example Features..." *Arduino Playground*. N.p., n.d. Web. 04 Mar. 2014.  
<<http://playground.arduino.cc/Code/VisualMicro#UxV-24UVLC8>>.
3. "Micromouse - Maze Solver - Theory." *Micromouse - Maze Solver - Theory*. N.p., n.d. Web. 03 Mar. 2014.  
<[http://www.societyofrobots.com/member\\_tutorials/book/export/html/94](http://www.societyofrobots.com/member_tutorials/book/export/html/94)>.
4. "Micromouse." *Wikipedia*. Wikimedia Foundation, 02 Sept. 2014. Web. 03 Mar. 2014. <<http://en.wikipedia.org/wiki/Micromouse>>.
5. "The Rat Pack." *Home*. SDSU, n.d. Web. 03 Mar. 2014.  
<<http://ratpack.sdsu.edu/>>.

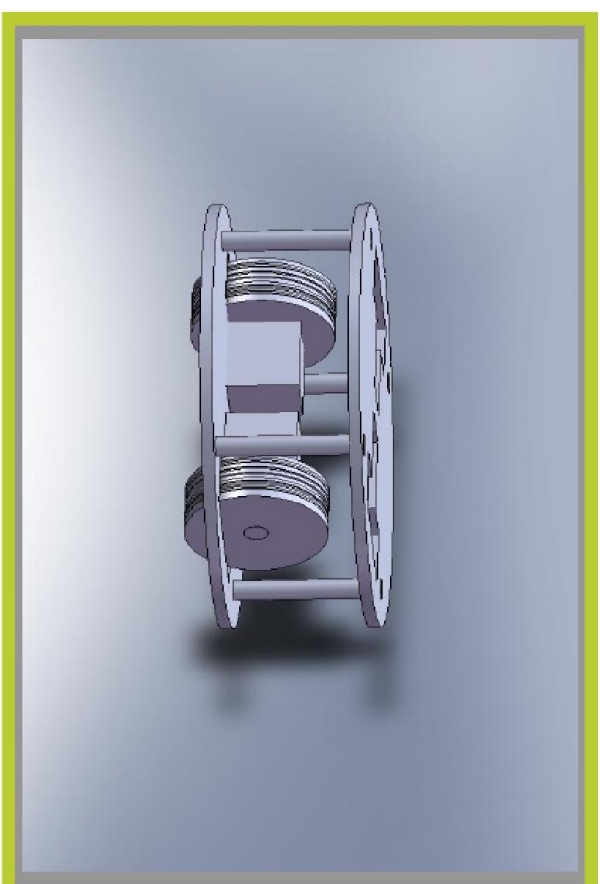


# Micromouse

## By: DeadMou5

### Design Team Mission

DeadMou5, the intelligent group of engineers from the highly accredited establishment of SDSU, will create an autonomous robot for the micromouse competition. The robot will be equipped with ultrasonic sensors, dc motors, and a signal receiving IR sensor to find its path through an unknown maze and retrieve a message from the beacon to return back to the start of the maze. The goal is to design the fastest and most efficient robot that will prosper on the day of competition!



3D Concept Model for Micromouse

- ### Engineers
- Jeff McGarran
  - Willy Herrera
  - Sami Elias
  - Evan Jina
- ### Sponsor
- Professor Gordon Lee

### Micromouse Background

Micromouse is an event where small mechanical robots compete in finding the center of a maze. The robots must be completely autonomous. Each maze consists of 16x16 cells with each cell at 18x18 cm<sup>2</sup>. The design of the robot usually is equipped with sensors for navigation and wall sensing as well as a clever algorithm. The algorithm stores all cells that are visited. This establishes a map of the maze and therefore once the center point is reached, the shortest route to the starting position can be easily retrieved. It is a very open competition which allows many modifications to both hardware and software yet it remains challenging to achieve an efficient system which can satisfy all constraints.