Maze Project

Mission/Game Objective

Your team has been tasked with designing a self-balancing robot that will be participating in a competition to map an unexplored maze/cavern/ruin and then retrieve all of the valuables and weapons in the shortest time possible. There will be monsters blocking certain paths and they can be eliminated with certain equipment that can be picked up within the maze. The expedition will be done on a unique maze/cavern/ruin that is generated by randomly placing markers (valuables, monsters, weapons), walls, and bridges on the printed maze. The robot will have a designated starting position that may or may not be at a maze entrance and it will need to return to that location with the valuables/loot. It is anticipated that teams will use the mapping phase to discover the location of the wall(s), bridge(s), and marker(s) within the maze, prior to the retrieval phase.

Position Descriptions

Navigation Design Engineer (Hardware & Software)

The navigation engineer is responsible for designing and testing the hardware and software needed to maneuver the robot within the maze and obtain information about its current surroundings. This can be broken down into the following key objectives

- Solution for Path Following (staying within the boundaries of the room and moving to the next one)
- Solution for Room Detection (hardware to detect walls and bridges AND/OR software agent)

During the software development phase of the project, the provided IR sensors may be used to help start the development of the **path following (**outer IR sensors**) and room detection (**inner IR sensors or software agent**)** algorithms. As part of the mission, each team may choose a method/sensor(s) for path following and room detection. The solution chosen may not use the short-range IR sensors included with the provided IR forward sensor shield.

<u>Ideas</u>

- Possible sensors that could be employed for path following and/or room detection of the provided basic maze include camera(s), IR, LDC, color, etc.
- Possible passive or active devices that could be added to the paper maze to create a smart maze include embedded barcodes, qr codes, conductive ink, copper tape, etc. Here a different set of sensors would be used for path following and/or room detection.

- You may want to consider triangulation and a software agent for path following and room detection. Triangulation sensor solutions include ultrasonic “pingers'‘, ToF, Lidar, DWM1000, to name just a few. A software agent would then use the x-y coordinates returned by the sensors, with a firmware encoded version of the maze, to answer questions regarding the robot's immediate surroundings allowing path following (distance to walls) and room detection (is there a wall to my right, how close am I to the wall on my right) only. The software agent may not provide information not available to a robot navigating the maze using in situ empirically obtained information (basic and smart maze).
- Ultimately, your solution may use a combination of the above ideas or a total unimagined solution (think outside the box).

Card Reader Design Engineer

The card reader engineer is responsible for designing and testing the hardware and software needed to detect, identify, and record data encoded on the markers/cards placed on the paper maze. The solution used will need to be able to do the following

- Detect when the robot has encountered or gone over a marker
- Identify and record which of the three possible types of marker it is (valuables, monster, weapon)

During the software development phase of the project, “stub” software routines or manual input may be used for inputting marker data. For example you could add a barcode to the card and then use the inner IR sensors on the provided IR shield to return the marker type.

Ideas

- Possible design solutions that could be employed for encoding and reading card data include RFID, e-ink, compass, morse code, neural network, camera, IR, LDC, color, barcodes, QR codes, conductive ink, copper tape, etc.
- Ultimately, your solution may use a combination of the above ideas for marking the card and reading the data or a total unimagined solution.

Control Design Engineer

The control design engineer is responsible for designing and testing the hardware and software needed to keep the robot balanced as it drives around the maze on two wheels (caster wheel is removed). Any technique may be employed to drive on two wheels as long as the chassis does not touch the ground in some way.

Ideas

- Possible design solutions include using an IMU, accelerometer, or a sensor that can measure short millimeter distances.

During the software development phase of the project, the provided caster may be used to keep the robot upright.

Game Software Engineer

The game software engineer is responsible for developing the software that will handle the gameplay aspects of the mission. That involves the following

- Algorithm for exploring the maze during the mapping phase
- Updating the pre-encoded maze based on the location of placed walls and bridges.
- Program that will determine the optimal path to collect all of the markers in the shortest time possible
- Integration of room and card information provided by other teammates into optimal path solution

Design Note

Each design engineer will be responsible for designing, breadboarding, testing, and doing a PCB layout for their hardware element (path following, maze navigation, card reader, and control). The 3DoT supports a 16-pin top shield and forward 8-pin sensor shield. If these shields become a shared resource, one of the design engineers will be tasked by the project manager to integrate the two layouts into an “integrated” shield design.

It should further be noted that these two *shields (top and forward sensor) may not be connected electrically*.

How to Gain a Competitive Edge

To gain a competitive edge teams need not share their implementation solutions. Each team must implement a unique solution to each problem. A team will be informed if their proposed technique/solution is already being implemented by another team. The first team to submit a solution as discussed in class takes precedent. Do not forget that Beachboard submissions are time stamped.