EEE3099S Milestone 2

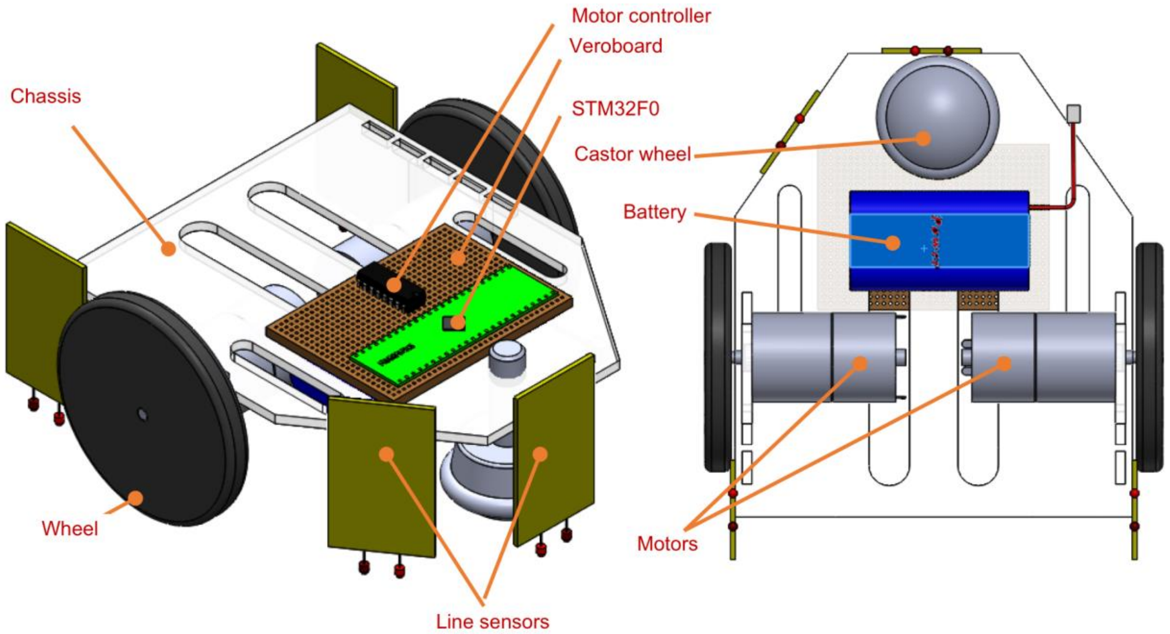
Group Number	5
Date	22/08/2019
Aronson, Jamie	ARNJAM004
Mehta, Ronak	MHTRON001
Martins, Danilo	MRTDAN014
Jericevich, Ricky	JRCRIC001

0 WORK DISTRIBUTION

Table 1 Work distribution for Milestone 2

Task Description	**Person(s) of Responsibility**
Shortest path Algorithm	ARNJAM004
Maze mapping algorithm	MRTDAN014
Flow chart design	ARNJAM004, MRTDAN014
Bill of Materials	MHTRON001
Sensor Design and physical sensor building	MHTRON001
Sensor description and circuit explanation	MHTRON001
Writeup	ARNJAM004, MRTDAN014, JRCRIC001, MHTRON001
Sensor simulation	JRCRIC001

1 SENSOR 1 INTRODUCTION



1.1 AIM

The aim of this sensor is for it to detect a black line and output acceptable TTL gate levels. If the sensor detects a black line, it should output a constant voltage high of between 2.8V and 3.3V. If no black line is detected, an output low should be displayed of between 0V and 0.5V.

1.2 SENSOR 1 USER REQUIREMENTS

The user requires that the sensor can detect a black line and outputting a voltage high when it does. The user also requires that the sensor be set up in a meaningful configuration such that an appropriate algorithm can be implemented to execute the desired task.

1.3 SENSOR 1 TECHNICAL REQUIREMENTS

Technical requirements include that an output voltage of between 2.8V and 3.3V be displayed when a black line is detected, as well as a 0V to 0.5V output when no black line is detected, thus a voltage low. It is also required that a supply voltage of between 7V and 12V be supported.

The sensor must have a single trigger event with digital output. It must also be capable of sensing the line accurately from 5mm to 30mm.

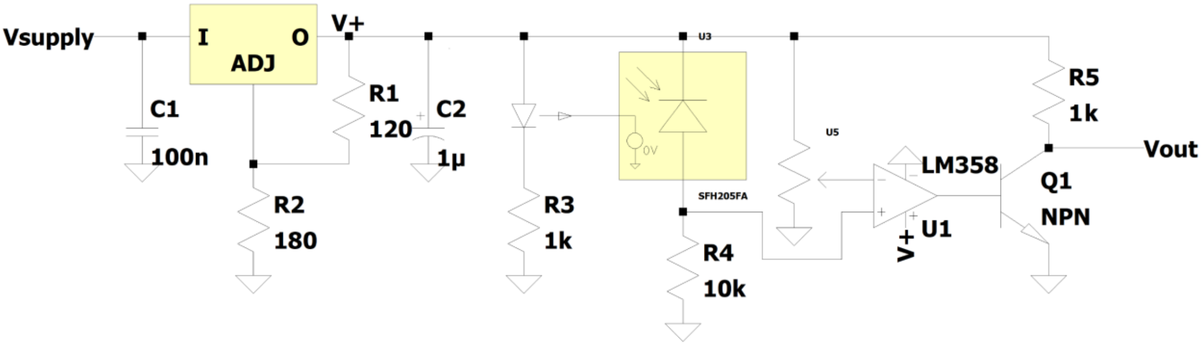
It should also be able to operate under no external lighting conditions, as well as normal day light conditions. All experiments and outcomes should be repeatable.

The sensor circuit must fit on a 30 mm x 30 mm Veroboard and use a 3 pin molex connector for Vcc, GND and the TTL digital output signal.

1.4 ALGORITHM 1 REQUIREMENTS

The algorithm we will use requires that the robot be able to map out an entire maze. Once this has happened, it is required that the robot be able to calculate the shortest path to the end of the maze from the same position that it started at. The algorithm should take into account anomalies in the maze such as loops, T-junctions, simple turns as well as 4-way crossings and any other intersection the robot may encounter.

2 SENSOR 1 (LINE SENSOR) CIRCUITRY



2.1 DESCRIPTION AND CIRCUIT EXPLANATION

The sensor circuit was used to detect the black line for the maze. It consisted of four stages in total, namely the regulating stage, sensing stage, buffering and comparing, and the level shifting stage.

LM317 adjustable voltage regulator was used for various input voltage between 7V and 12V. The output voltage was regulated to be at 3.1V using a combination of resistor ratios as per the datasheet. This output voltage was used to power the IR LED, IR photodiode, Opamp, potentiometer and transistor (for level shifter).

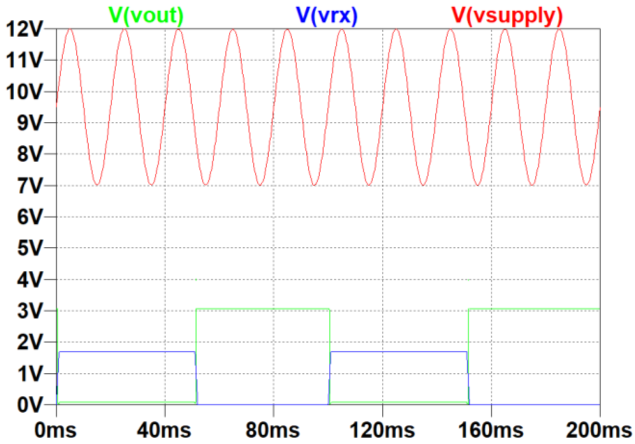
The sensing stage consisted of an Infrared LED and an IR photodiode. The IR LED emits infrared radiation which gets reflected when it hits a white surface and then gets received by the photodiode which converts the infrared radiation in terms of voltage. If the IR beam hits a black surface, then nothing is received by the photodiode and hence there will be no voltage drop across it.

For the buffer and comparing stage, LM358 was used as a comparator which has two inbuilt low noise opamps. Here, a potentiometer was used to set up reference voltage at the positive terminal of the opamp and the photodiode sensing voltage as the negative terminal input to the opamp. The comparator then compared both these voltages and outputs a digital signal which connects to the level shifter. The positive rail of the opamp was tied to the voltage regulator output and the negative rail to ground.

An NPN transistor implemented as a switch was used for the level shifter to output a constant digital voltage of 3.1V when high and 0V when low. The base of the transistor was connected to the opamp output, the collector was tied to voltage regulator output and the emitter to ground. A Red Led was also used to help detect for a digital high or low. It would light up if it got 3.1V from the output and turn off when it got close to 0V.

The output from this sensor circuit goes to the input of the STM for coding purposes.

2.2 Simulation Result and Discussion



As can be seen from the above image, the supply voltage (red) into the censor circuit varies between 7 V and 12 V. The IR photodiode's reaction to receiving and not receiving the IR light is simulated by the blue signal, which shows the signal at the input to the inverting terminal of the opamp. The output voltage of the sensor circuit is shown in green. The above image shows that despite a non-constant supply voltage, the output voltage remains a constant 3.1 V when high and approximately 0 V when low. Furthermore, the output voltage only goes high when the photodiode's output goes below approximately 0.8 V (no light is detected when over the black tape). When the photodiode's output is higher than 0.8, the output voltage goes low. Note that this threshold of 0.8 would be altered using the potentiometer as per the lighting conditions so that the sensor can detect the black tape in different lighting conditions.

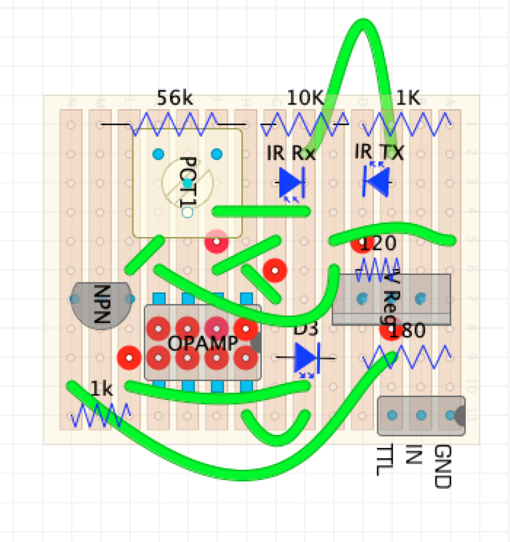
2.3 Bill of Material

Item #	Component	Quantity	Distributer	*Cost for @
1	SFH205FA Photodiode	1	White Lab	R11
2	TSAL6100 infrared LED	1	White Lab	R5
3	Red LED (1.8V forward voltage drop)	1	White Lab	R1
4	120Ω Resistor	1	White Lab	
5	180Ω Resistor	1	White Lab	
6	1kΩ Resistor	2	White Lab	
7	10kΩ Resistor	1	White Lab	
8	56kΩ Resistor	1	White Lab	
9	100nF capacitor	2	White Lab	
10	10kΩ Trimmer potentiometer (10% tolerance)	1	White Lab	R4
11	LM317 regulator	1	White Lab	R5
12	LM358 Opamp	1	White Lab	R4
13	PN2222A NPN Transistor	1	White Lab	
				R30

*Asked Brandon about component prices

3 Sensor 1 Building and Testing

3.1 Veroboard Design



3.2 Testing Procedure

Supply a sweeping voltage of 7V-12V to the input of the circuit and check to see that the output falls between 2.8V-3.3V when the sensor detects a black line and 0V when no black line is detected.

Place the sensor 5mm away from the black line and check to see that the correct output voltage is given. Then put the sensor 30mm away from the black line and check that the correct output is given.

Place the sensor under different lighting conditions and check to see that the correct output voltage is given for each case.

3.3 Result

The result was that the sensor confirmed to all of the tests undergone and that the output voltage was approximately a constant of 3.1V for black line detection and a voltage of 0.2V for the white spaces.

3.4 Conclusion

The sensor has been built in accordance with the technical and user requirements and will perform the necessary tasks to satisfaction. An appropriate output voltage of 3.1V has been achieved using a level shifter and voltage regulator.

4 ALGORITHM 1

4.1 APPROACH AND EXPLANATION

There are two algorithms required to meet the user requirements.

4.1.1 Maze mapping

The outcome of the algorithm is an undirected weighed graph structure containing all nodes (joints) and edges (paths connecting two joints) with their respective weights (time taken to travel the edge, in this case) which will be used to find the shortest path.

The robot starts by calibrating itself. Once it's aligned with the line, it names that direction "North". This will be fixed. It then goes to the first node it finds and sets it as the starting position. It adds the position to the an array containing the coordinates (time_x, time_y)[1].

Every time it finds a new node[2] it appends it to the Nodes array. And every time it travels between two points, it adds an edge containing both nodes, as well as the time in between them.

It also has an array with the types of each node (dead end, turn with 1 exit, 2 or 3 exit junction). When a new node is found, its type is added to the array. In addition, it has an array of explored edges of visited nodes. For each node, it keeps track of which exits have been used, so the robot gives preference to unexplored exits. When the number of exits explored for each node is not less than its type, the robot knows is has mapped the whole maze, and therefore, goes to the black circle, which should had been found during maze mapping

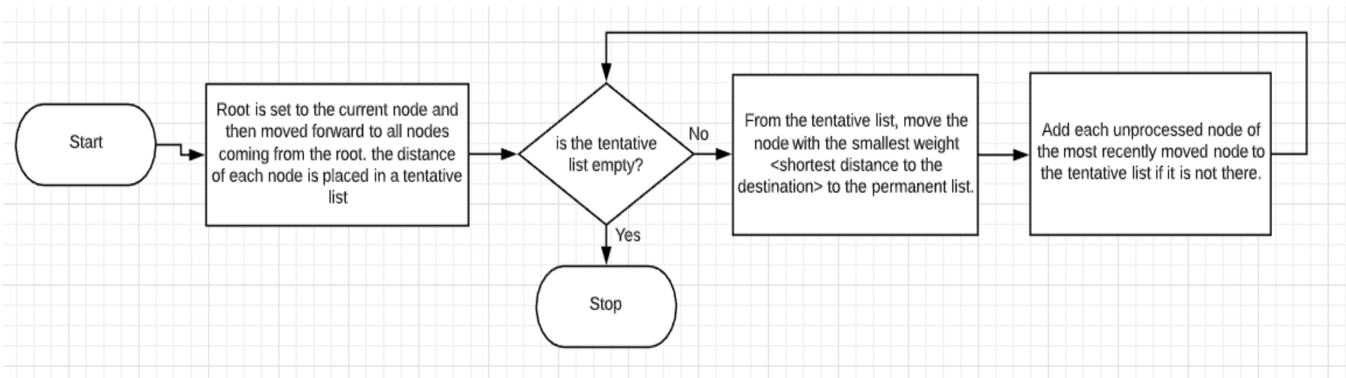
4.1.2 Shortest path

With the whole maze having been mapped, Dijktra's algorithm [1] can be used to find the shortest path.

The algorithm considers intersections as nodes. Each possible route from a base route is then considered. The path that is closest to the destination is the one that is taken. The process of setting a new base node and then analyzing possible paths is then repeated until no possible paths are available, i.e. the robot is at the end of the maze.

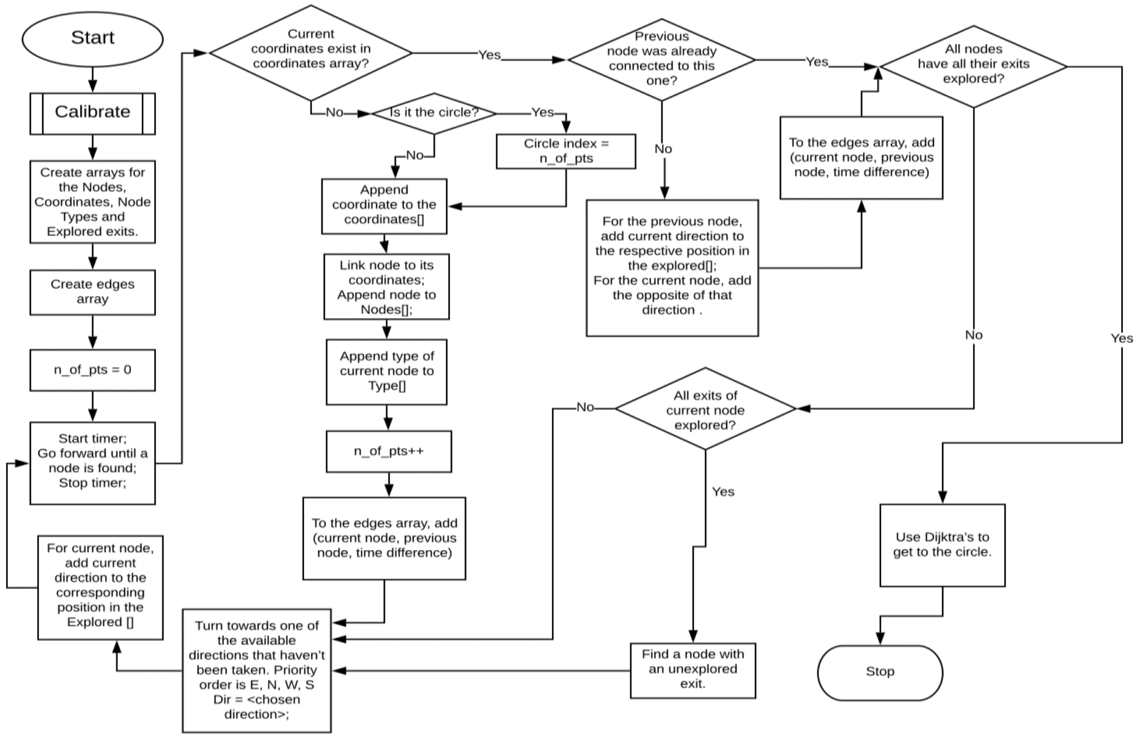
4.2 FLOWCHART

4.2.1 Shortest path algorithm



[1] Traveling East would increase time_x, and travelling West would decrease it. Going North would increase time_y, and South, decrease it.
[2] Found by comparing with coordinates of known nodes.

4.2.2 Maze mapping flowchart



5 BIBLIOGRAPHY

[1] HTTPS://WWW.GEEKSFORGEEKS.ORG/DIJKSTRAS-SHORTEST-PATH-ALGORITHM-GREEDY-ALGO-7/

[2] HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/6997314

[3] HTTPS://EMBEDJOURNAL.COM/SHORTEST-PATH-LINE-FOLLOWER-ROBOT-LOGIC-REVEALED/

[4] https://www.geeksforgeeks.org/mathematics-graph-theory-basics-set-1/