On the Design of Maze Wanderer Robot

Mohanad Abdulhamid[1,a)]

[1] Al-Hikma University, Iraq

[a)] moh1hamid@yahoo.com

Abstract. This paper focuses on designing, programming and implementing a maze wonderer robot that takes the commands to dictate its direction of motion based on the obstacle avoidance. A program is to be written to give the robot its intelligence. The paper will as well be restricted to a motorized car which will be given the intelligence to be able to navigate through a given maze.

INTRODUCTION

Robots are intelligent machines capable of doing tasks they are programmed to do. They have shown significance in decreasing human work load especially in industries. If there is one technological advancement that would certainly make living easy and convenient, robots would be the answer.

Robots are mostly utilized in the manufacturing industry. People who do the same thing for a long period of time tend to get bored and tired of what they are doing and might arrive in a position wherein they are unwillingly doing their job. The person who reached this point will not be as efficient and effective as when they first started working. Also, as human beings, we get exhausted so the length of time that we can work is limited. This is when the importance of robots is realized. They can be set to function for a long span of time producing the same quality product all throughout the production process. This results to an increase in the number of manufactured products of consistent quality and decrease in the production of defective goods.

Industries can gain a lot of benefits out of robotics. The company productivity will rise making businesses achieve more profits. Also, company losses will be reduced because flawed products are trimmed down to almost none. The importance of automation and robotics in all manufacturing industries is growing. Robots can replace human beings in a wide variety of industries. Robots outperform humans in jobs that require precision, speed, endurance and reliability. Robots safely perform dirty and dangerous jobs. Robots need no environmental comforts as compared to humans and can process multiple tasks simultaneously.

For repetitive tasks which use the same path in a factory to remove the need for a human operator, path can be used as a guide for a robot lawnmower. Smarter versions of path followers can be used to deliver mail within an office building and deliver medications in a hospital. The technology has been suggested for running mass transit systems within a factory/industry and may end up as part of autonomous cars navigating the freeway.

Rescue robots in development are being made with abilities such as searching, reconnaissance and mapping, removing rubble, delivery of supplies such as medical supplies or even evacuations of casualties.

From the above uses one can be convinced that indeed a robot is a very important device in our day-to-day operations. Thus, proper design and implementation of this instrument, taking into consideration safety, accuracy and precision, cost and efficiency has been an engineering concern to meet the ever growing demand of the device. Some works on the topic of robot design can be found in literatures[1-6].

IMPLEMENTATION METHODOLOGY

Hardware development

The hardware part must be correctly design to ensure that the operation will work appropriately as desired. The main operation is to define how to control the DC motors using the H-bridge and the sensor and how to connect this combined circuit to the micro-controller circuit. The H- bridge and the micro-controller are being controlled by the micro-controller program code.

The main circuit consists of a combination of three circuits; the power supply unit, sensory array module, central processing unit (controller) and the drive system which is the motor control unit, all are discussed in the next sections.

Central processing unit (controller)

Control unit refers to an electronic system which takes inputs from the various sensors which collect data from the environment and can drive the output devices according to the conditions which are applied due to various constraints. The control unit consists of a programmable logic device called microcontroller. The microcontroller is a type of electronic device which can be pre-programmed according to our requirements. Every microcontroller has various input and output pins where different I/O devices can be connected. The microcontroller also has other peripherals embedded inside the same chip. Therefore microcontroller is nothing but a microprocessor with all other peripherals embedded inside the same chip. Whenever we have to control the systems dynamically according to the conditions of system environment, we use a microcontroller as the control unit.

Design of the obstacle sensing unit

Each infrared range sensor measures the distance to an object by detecting reflected infrared light transmitted by its light emitter. This is illustrated in Fig.1. The electronics in the sensor enable it to measure the angle at which the reflected light enters the detector. When the sensor is close to an object, light enters the detector at a sharp angle. When the sensor is far from an object, light enters the detector at a slight angle. The sensor outputs an analog voltage that varies depending on the angle at which the reflected light enters the detector. This technique makes the sensor insensitive to ambient light and the reflectivity of the detected object, ensuring the output voltage is solely a function of the distance to the detected object.

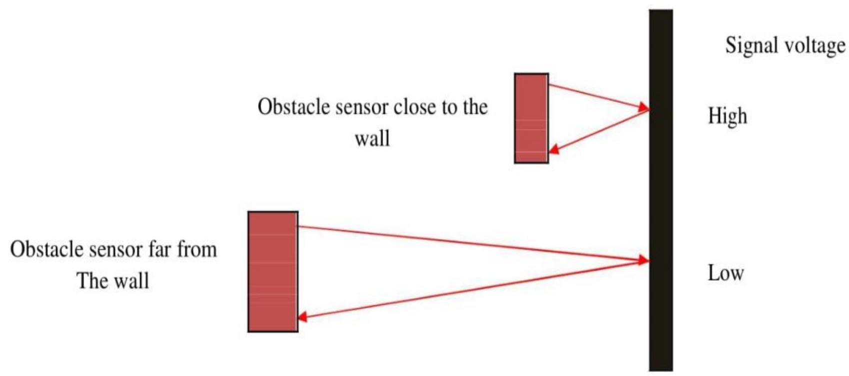

FIGURE 1. Sensing Distance to the Obstacle

The infrared based object detector can be implemented using two configurations; break beam sensors and reflectance sensors. The second configuration is very suitable for the portability of this robot where both the infrared(IR) source and the IR detector need to be on the robot; hence it is going to be implemented in the control system of the robot. The infrared emitter-detector basic circuit is as shown in Fig.2.

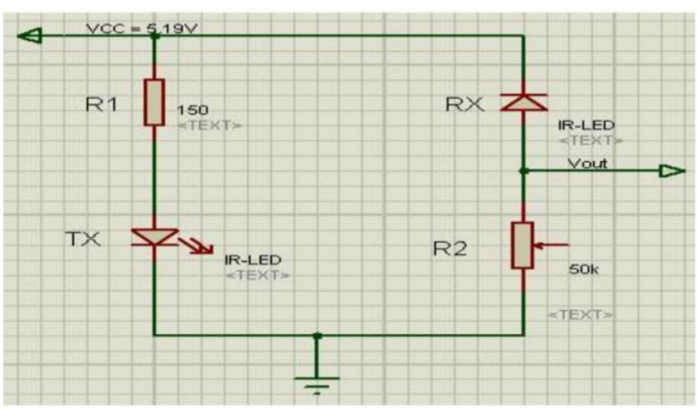

FIGURE 2. Infrared emitter-detector basic circuit

R1 is chosen to be 220Ω to prevent the light emitting diode(LED) from melting itself. The resistance value of R2 determines the sensitivity of the robot in terms of the distance between the robot and the obstacle. For R1 = 220Ω, R2 = 22KΩ and Vcc = 5.19V, if no obstacle is in front of the sensor then the value of Vout is around 2.05V and when an obstacle comes to about 15cm from it, the value goes up to 3.78V, and when the obstacle comes nearer than 5cm the value saturates to 5.06V.

Similarly, six units of the obstacle detector circuits designed above are going to be used, two for detecting left wall obstacles during the forward motion, two for detecting front obstacles and turning right if there is an obstacle in front. One on the right to determine when to move forward when there a reverse initially and one on the rear part in order to turn when reversing.

All the outputs from the sensors are analogue in nature and since the microcontroller to be used has only six pins that can do analogue to digital conversion. In addition to the microcontroller that is going to do the analogue to digital conversion, it also performs the signal processing.

Design of the control signals processing unit

Output signals from the obstacle detection unit are used as inputs to the control signals processing unit. These include the six inputs from the other six inputs from the obstacle detector circuits. Hence the control signals processing unit will have six inputs from the obstacle detection unit which are all digital in nature.

Since the input signals are analogue signals, the first step is to convert them to digital signals using the analogue to digital conversion feature of the microcontroller. Next step is to perform logical operations on the digital signals; for the obstacle detection circuits, the one with the highest value (meaning receiving the highest value due to complete obstability) is determined to indicate the direction of motion of the robot and for the obstacle sensors, their values are used to determine whether the motion to a given direction can be permitted.

The robot will have a total of six IR sensors, all oriented strategically to determine the motion of the robot. Orientation of the sensors is as shown in Fig.3.

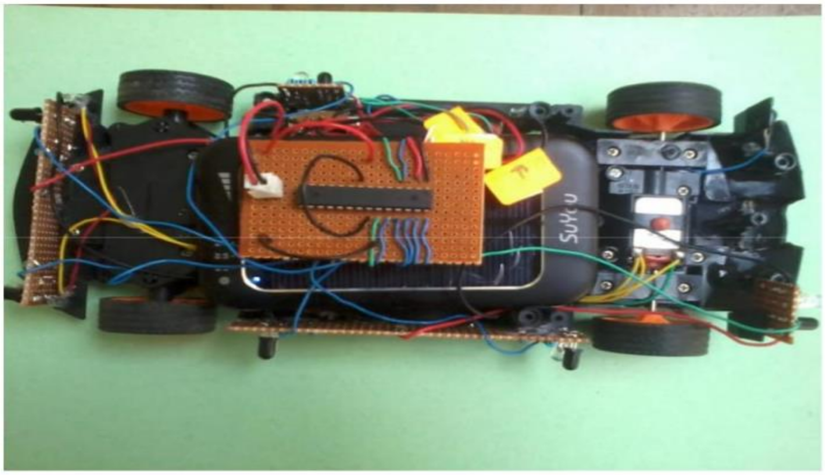

FIGURE 3. Orientation of the IR infrared sensors

The total number of sensors that the robot is expected to use are six. This implies that there will be a number of combinations as the various sensors will be giving different readings at any instant of time.

Software development

The primary purpose of the software is to maintain control over the hardware at all times and determine where to move by solving the maze. Controlling the hardware consist of reading the sensors; setting the motor speed and communicating with any external peripherals

Algorithm

The principal goal of a robot is to solve the maze and find its end. To accomplish this task, the robot uses a particular maze searching algorithm. A vast amount of research on searching techniques already exists and is currently being undertaken. As a result, robots generally use some variation of the following three searching algorithms: wall follower, depth first search and flood fill. Only, wall follower algorithm is considered here.

Wall follower algorithm

The wall follower, the best-known rule for traversing mazes, is also known as either the left-hand rule or the right-hand rule. If the maze is simply connected, that is, all its walls are connected together or to the maze's outer boundary, then by keeping one hand in contact with one wall of the maze, the robot is guaranteed not to get lost and will reach a different exit; otherwise, the robot will return to the entrance having traversed every corridor in the maze at least once.

Another perspective into why wall following works, is topological. If the walls are connected, then they may be deformed into a loop or circle. Then wall following reduces to walking around a circle from start to finish. To further this idea, by grouping together connected components of the maze walls, the boundaries between these are precisely the solutions, even if there is more than one solution.

If the maze is not simply connected (if the start or endpoints are in the center of the structure or the pathways cross over and under each other), this method will not be guaranteed to help the goal to be reached.

Wall-following can be done in 3D or higher-dimensional mazes if its higher-dimensional passages can be projected onto the 2D plane in a deterministic manner. However, unlike in 2D, this requires that the current orientation be known, to determine which direction is the first on the left or right.

Fig.4 below shows a simple maze which it was use for the basis of illustration how the robot will navigate around it until it exits.

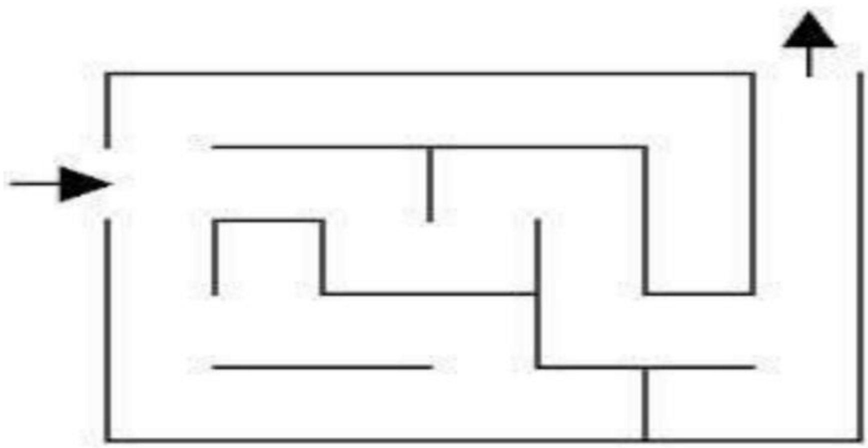

FIGURE 4. Simple Maze for wall follower algorithm

The final implementation was done with the complete system incorporating sensors, the central processing unit and the power supply then couple with the radio controlled toy car was finally done as shown in Fig.5 below.

FIGURE 5. Final implementation

Programming environment

Arduino is a single-board microcontroller, intended to make the application of interactive objects or environments more accessible. The hardware consists of an open-source hardware board designed around an 8-bit Atmel AVR microcontroller, or a 32-bit Atmel ARM. Current models feature a USB interface, 6 analog input pins, as well as 14 digital I/O pins which allow the user to attach various extension boards.

It comes with a simple integrated development environment (IDE) that runs on regular personal computers and allows writing programs for Arduino using C or C++ to the atmega 328 then the chip was transferred to the robot's board. The Fig.6 and Fig.7 below show the Arduino Uno board and the pin mapping for the chip.

FIGURE 6. Arduino Uno Board

FIGURE 7. Atmega pin mapping

SIMULATION RESULT

Simulation of the obstacle sensors

Six obstacle sensors were used to detect the presence of an obstacle in the path of the robot. All of the sensors were similar (they had approximately equal output voltage for a given distance from the obstacle) hence they were having the same response. Fig.8 is the circuit diagram of the obstacle detector designed for the robot.

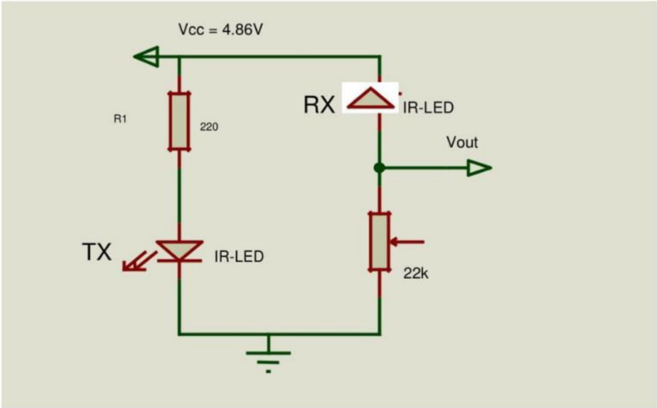

FIGURE 8. Obstacle detector circuit diagram

For Vcc = 4.86V, R1 = 220Ω and R2 = 22KΩ, the obstacle was moved in a straight line along the direction of the obstacle sensor. The corresponding values of the distance of the obstacle from the obstacle sensor (d), and the obstacle sensor's output voltage (Vout) were measured and recorded in table 1.

TABLE 1. Result of the obstacle sensors

Number	Distance, d (cm)	Output Voltage (Volts)
0	No obstacle	1.73
1	35	2.30
2	30	2.41
3	25	2.57
4	20	2.77
5	15	3.03
6	10	3.44
7	5	4.12
8	4	4.35
9	3	4.53
10	2	4.69
11	1	4.70

The plot of the output voltage of the obstacle sensor in volts against the distance of the light source from the sensor in centimeters is as shown in Fig.9.

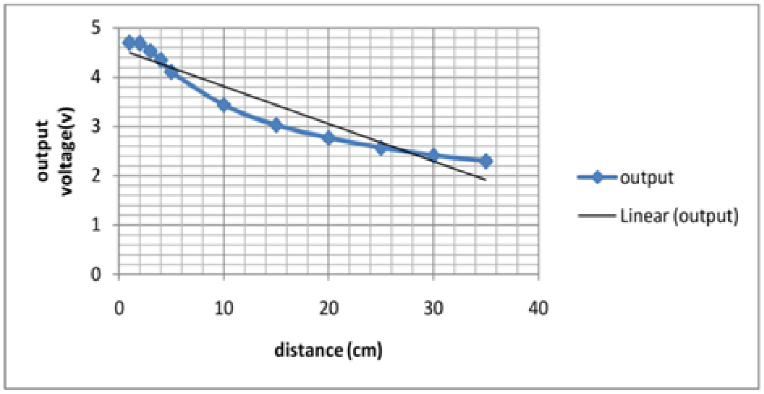

FIGURE 8. Variation of voltage with distance for obstacle sensors

From the plot of the output voltage of the obstacle sensor in volts against the distance of the light source from the sensor in centimeters, it is seen that there is a 1/d^2 variation of distance and the output voltage, with d (cm) being the source distance from the sensor. Therefore as the distance of the obstacle (in the path of the robot) from the sensor decreases the output voltage also increases.

Simulation of the signal processing unit

Output signals from the obstacle detection unit are the inputs to the control signals processing unit. These include the six inputs from the obstacle detectors, which are digital in nature. This implies that the control signals processing unit has six inputs of which are digital in nature.

The test done on the signal processing unit where each one of all of the input signals from all the sensors gave particular instructions as the output of the signal processing unit. Indicated below in table 2 are the various possible states of the robot and the corresponding control action is also suggested.

TABLE 2. States of the robot and the corresponding instructions

Number	Sensor	Instruction
1	Left _forward sensor	Move forward
2	Left _reverse sensor	Move forward with a left turn if left _forward sensor is clear.
3	Forward _left sensor	Move forward with a right turn if the left _forward is blocked
4	Forward _right sensor	Move forward with a right turn if it's not clear, else reverse with left turn until it's clear if all other options are not clear
5	Reverse sensor	Move reverse with left turn until forward right is clear.
6	Right _forward sensor	Move forward with right turn if it's clear and front and left not clear

DISCUSSION

The accuracy value is the most important parameter when conducting a study to detect diarrheal disease because True Positive and True Negative are parameters that will be used as a diagnosis for further treatment. If the accuracy value is too low, the model does not have the ability to distinguish between right and wrong objects, so there is no advantage and will cause harm to sufferers. Model 3 has an accuracy value of 92%, so model 3 is the model that has the highest accuracy value among the other models.

The recall value is the second important parameter because it is the value of the comparison between True Positive and all true actual data. In conducting a study on the detection of diarrhea, the appearance of False Negatives was not expected. False negatives or Escherichia coli bacteria that are negatively detected will be detrimental to the patient because it causes the bacteria to be considered as not Escherichia coli bacteria or considered as another object so that the patient will be considered fine by the doctor. The relationship between recall values and False Negatives in equation 2.12 is inversely proportional, so that the fewer the number of False Negatives, the higher the recall value. Model 3 has a recall value of 95%, so model 3 is the model that has the highest recall value among the other models.

The precision value is an important parameter number three because it is the value of the comparison between the True Positive and all the detection results that the model considers true. In accordance with equation 2.11, if there are fewer False Positives, the precision value will be greater. Even if it is a False Positive or other object that is considered to have a large number of Escherichia coli bacteria, it will actually not be detrimental for sufferers of diarrhea because the patient will immediately be given further treatment by a doctor. Model 3 has a precision value of 96%, so model 3 is the model that has the highest precision value among the other models. and recall, which will be used as an option when the difference between the precision and recall values is too great.

CONCLUSION

In this paper, a maze wanderer robot was designed and implemented using wall follower algorithm technique. Controlling the movement of a robot is necessary for almost all type of robots, thus the device serves as a basic necessity accomplished while designing any high level robots. It was also established that wall follower algorithm has precise control, fast processing, reduced error rate and the most important being cost-effective as compared to depth-first and flood fill algorithms.

REFERENCES

1. K. Maritim, " Maze wanderer robot", Graduation Project, University of Nairobi, Kenya, 2014.
2. A. Kasiman," Optimization algorithm of autonomous maze solving robot", M.Sc. Thesis, Universiti Tun Hussein Onn, Malaysia, 2015.
3. M. Rahman, " Autonomous maze solving robot", Graduation Project, University of Liberal Arts Bangladesh, Bangladesh, 2017.
4. R. Kumar, " Maze solving robot with automated obstacle avoidance", Procedia Computer Science, Vol.105, pp.57-61, 2017.
5. A. Quasmi," Maze-solving robot", VJER-Vishwakarma Journal of Engineering Research, Vol.2, Issue 3, pp. 117-123, 2018.
6. S. Herbie, " Development of a maze solving mobile robot capable of tracking the distance it traversed", Advanced Science Letters, Vol.24, No.11, pp. 8640-864, 2018.