Abstract

For various robotic applications, our daily life is becoming smarter and handy day by day. The patients in the hospitals receive the medicine manually by the nurse/ward boy which is very trivial and inefficient. So, there is a need for the automation in this area. Our idea is to build SmartMed (robot) which delivers medicine to the patient on the right time without fail. This reduces the possibility of patients getting wrong medicines and also reduces man power in hospital. The device is build using Raspberry Pi (System on Chip SoC), Infrared sensors, LD293 driver IC, stepper motors etc. The IR sensors will be used to follow the path and the LD293 driver IC is used for to control the stepper motor rotations. The device will work on network which will be used to deliver medicine to each and every patient in the hospital. Our experimental results shows that we have achieved 80% of accuracy.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Description
IR	Infrared ray sensor
GPIO	General purpose input output
VCC	Voltage collector to collector

Chapter 1

Introduction and Objectives

1.1 Introduction

In many government hospitals, most of the times patients do not get their medicines on time, because of various reasons like negligence, undersized staff which leads to the deterioration in their health condition. And there is no efficient existing system that can resolve this problem, which made us to think and get to this solution. For various robotic applications, our daily life is becoming smarter and handy day by day. The patients in the hospitals receive the medicine manually by the nurse/ward boy which is very trivial and inefficient. So, there is a need for the automation in this area. Our idea is to build SmartMed (robot) which delivers medicine to the patient on the right time without fail. This reduces the possibility of patients getting wrong medicines and also reduces man power in hospital. The device will work on network which will be used to deliver medicine to each and every patient in the hospital. The device is build using Raspberry Pi (System on Chip SoC), IR sensors, LD293 driver IC etc.

1.2 Objectives

1) The system must deliver the medicine to the patients on time.
2) The system must be scalable, reliable and cost effective.
3) The system must reduce the interaction between the patients and nurses/ward boys.

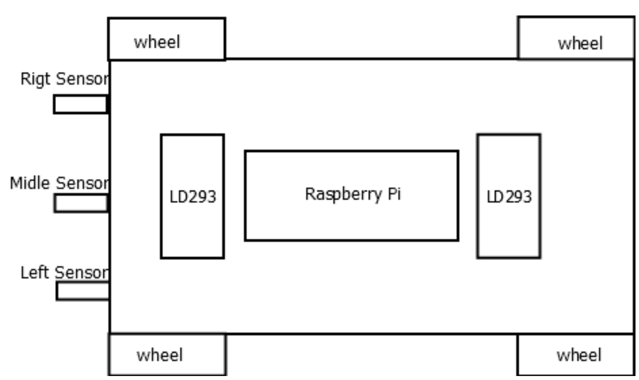

Fig.1 Block diagram

Chapter 2

Literature Review and Research Gap

2.1 Literature survey

A recent study of Johns Hopkins claims more than 250,000 people die every year from medical errors. Other reports claim the numbers to be as high as 440,000 medical errors are the third-leading cause of death after heart disease and cancer. Advocates are fighting back, pushing for greater legislation for patient safety.

2.1.1 Journal Papers 1

Title: Design and implementation of static and dynamic objects store systems using line follower robots.

In [1], this paper introduces static and dynamic methods for objects store system using the line follower robot. The static means the robot moves on static lines to reach to any store location while the dynamic means that the arranged of the following lines is changed according to the location of the storage box. The static objects store system is represented by the Digital differential algorithm DDA and the dynamic objects store system is represented by the Bezier curve algorithm. In both environments the propose store system consists of several boxes that arranged in several columns. These boxes can be used to store and return small objects like mobile phones according to a secret code that entered from a user interface unit. The principle of store and restore of these objects is dependent on using line follower mobile robots. Both the methods are designed and software implemented in an environment with thirty boxes arranged in four columns. The comparisons between these algorithms are shown with respect to the length of the paths and time of arrival.[1]

2.1.2 Journal Papers 2

Title: Design and implementation of a robotic technique-based waiter.

In [2], robotic waiter is one of the best examples of daily life automation. By this technology, a mobile robot can take order and serve food or beverage to the person in any environment of restaurant, hotel, office or household. This paper presents description about a robotic waiter technique along with the design of microcontroller-based robot which works effectively in a restaurant and office environment within a certain area. A prototype design of a wheel based moveable robotic waiter has been shown in this paper. The prototype waiter robot has been designed and implemented to receive order-request via android apps, then to collect food/drinks(max600gram) from kitchen boy, then to travel to the destination (the person who send order-request) and again to come back to its source point completing the order of requested food/drinks**[**2].

.

2.2 Research Gap

- We have used same technology as from the journal paper 1 but the application differs, they are using it for the store system management and we are using it for the medicine delivery system.
- In the second journal paper they are using unique table numbers to identify the location of the customers where as in our system we use a network to deliver the medicine to patients.

Chapter 3

Prior-art search

3.1 Patents/copyrights 1

Title: (CN106873598) Laser line-following robot and line following method thereof.

(1) The invention provides a laser line-following robot and a line following method thereof.

(2) On the basis of the improved line following mechanism, a guiding line having chromatic aberration between the ground and the line is laid on the ground and is used as a line-following track and a laser line-following robot is used for realizing line following, so that autonomous tracking following and autonomous positioning of the robot can be realized with high sensitivity and stability.

Link:
https://patentscope.wipo.int/search/en/detail.jsf?docId=CN199691006&tab=NATIONALBIBLIO&office=&prevFilter=&sortOption=Relevance&queryString=EN_AB%3A%28line+follower+robot%29&recNum=1&maxRec=678

3.2 Patents/copyrights 2

Title: (IN2791/MUM/2012) "VURTUAL LINE FOLLOWER".

(1) Practically much preferred concept as path of these robots may have to be changed. But if the path is not physical, or say imaginary, the drawback of previous version can be overcome. This is the main concept of this project: "Virtual Line Follower".

(2) This is done by making that line virtual, means the line should be present in the robot's memory. So that it can be changed anytime with an ease.

Link:
https://patentscope.wipo.int/search/en/detail.jsf?docId=IN211608025&recNum=2&office=&queryString=EN_AB%3A%28line+follower+robot%29&prevFilter=&sortOption=Relevance&maxRec=678

3.3 Patents/copyrights 3

Title: (CN103830065) Medicine delivery robot based on T-type rail.

(1) The invention discloses a medicine delivery robot based on a T-type rail.

(2) Two schemes are included. According to the first scheme, the robot comprises the T-type rail, wheels, a transmission device, a robot support and a robot body. According to the second scheme, the robot comprises an inward concave rail, wheels, a transmission device, a robot support and a robot body.

Link:

https://patentscope.wipo.int/search/en/detail.jsf?docId=CN105267737&recNum=1&office=&queryString=EN_AB%3Amedicine+AND+EN_AB%3Adelivery+AND+EN_AB%3Arobot&prevFilter=&sortOption=Relevance&maxRec=24

3.4 Patents/copyrights 4

Title: (CN106426218) Hospital medicine delivery robot.

(1) The invention discloses a hospital medicine delivery robot.

(2) The hospital medicine delivery robot comprises a robot body; a medicine placing box is arranged on the robot body; a low-temperature placing area and a high-temperature placing area are arranged in the medicine placing box, and are separated through a separation layer; multiple placing plates are respectively arranged on the low-temperature placing area and the high-temperature placing area.

Link:

https://patentscope.wipo.int/search/en/detail.jsf?docId=CN192990338&recNum=3&office=&queryString=EN_AB%3Amedicine+AND+EN_AB%3Adelivery+AND+EN_AB%3Arobot&prevFilter=&sortOption=Relevance&maxRec=24

Chapter 4

System Requirements

4.1 Introduction

System requirements cover all of the requirements at the system level, which describe the functions of the SmartMED should fulfill to satisfy the requirements. It is expressed in an appropriate combination of textual statements, views and non-functional requirements the latter expressing the reliability, scalability and standards to meet.

4.2 Software Requirements

Table No.1 Software Requirements

Software	**Use**
Raspbian	Operating System
Python	Programming language to code and configure the bot
IDLE	IDE for python programming language
Putty	For Secure Shell (SSH) connection
Vim	Command Line Editor

4.3 Hardware Requirements

Table No.2 Hardware Requirements

Hardware	**Use**
Raspberry Pi	Single board computer for controlling the bot
LD293	Motor Driver IC

IR Sensors	For sensing the path
Jumper wires, Breadboard, Raspberry Pi T-Cobbler	For connection purpose
Stepper motor	For movement of wheels
Chassis materials	Material used to construct vehicle frames

4.4 Functional Requirements

1) Must deliver the medicines on time.

2)Must deliver the medicines to specified patients.

3) Must alert the patients by a sound message.

4.5 Non-functional requirements

1) Must be extremely reliable.

2) Must have a definite network.

3) Charging must be fast but battery should have long life.

4) Must use less power as much as possible.

5) Must be cost effective.

Chapter 5

System Analysis and Design

5.1 Introduction

Systems analysis is a problem-solving technique that decomposes a system into its component pieces for the purpose of the studying how well those component parts work and interact to accomplish their purpose. Systems design is the process of defining the architecture, components, modules, interfaces and data for a system to satisfy specified requirements. This chapter explains the System Design and Modules of the SmartMED.

5.2 Objectives of the system

1. The IR sensor module must sense the path and give it as input to the system.
2. The LD293 IC and stepper motor module must take input from the GPIO pins and control the wheels accordingly.

5.3 System Design

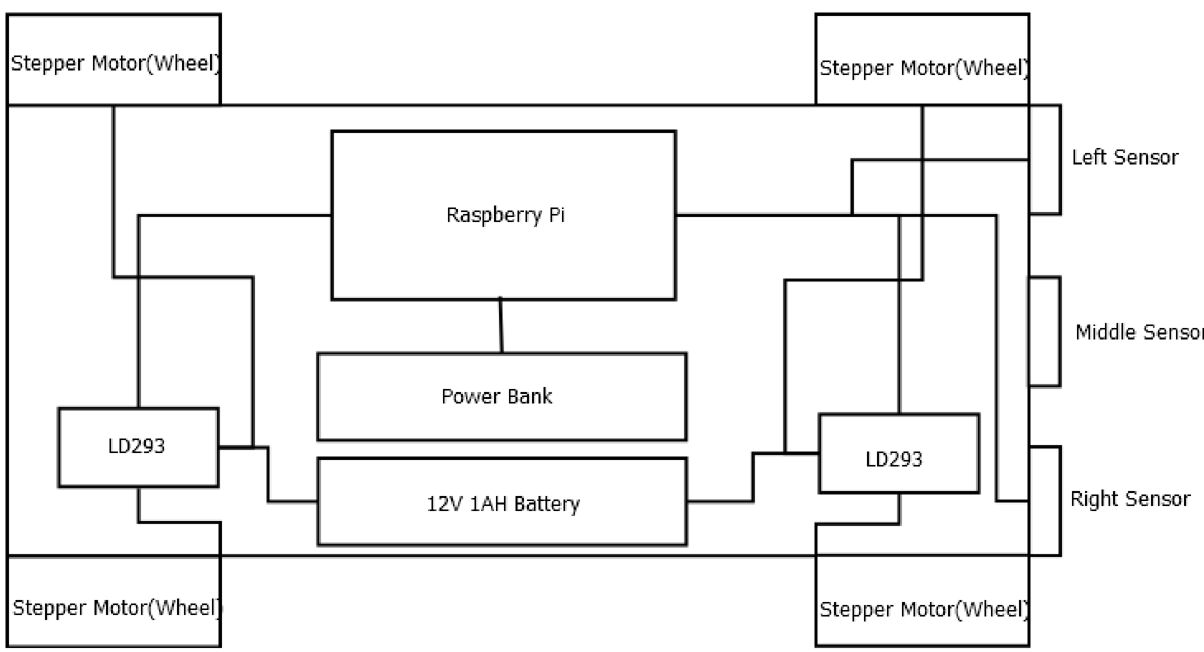

Fig.2 Component diagram

The fig 1 shows pictorial representation of the system. The system consists of Raspberry pi, LD293 IC, IR sensors and stepper motors. The device has IR proximity sensors for sensing purpose. LD293 IC is used for controlling the movement of the wheels. Raspberry pi and stepper motors are powered by power bank and 12V acidic battery respectively.

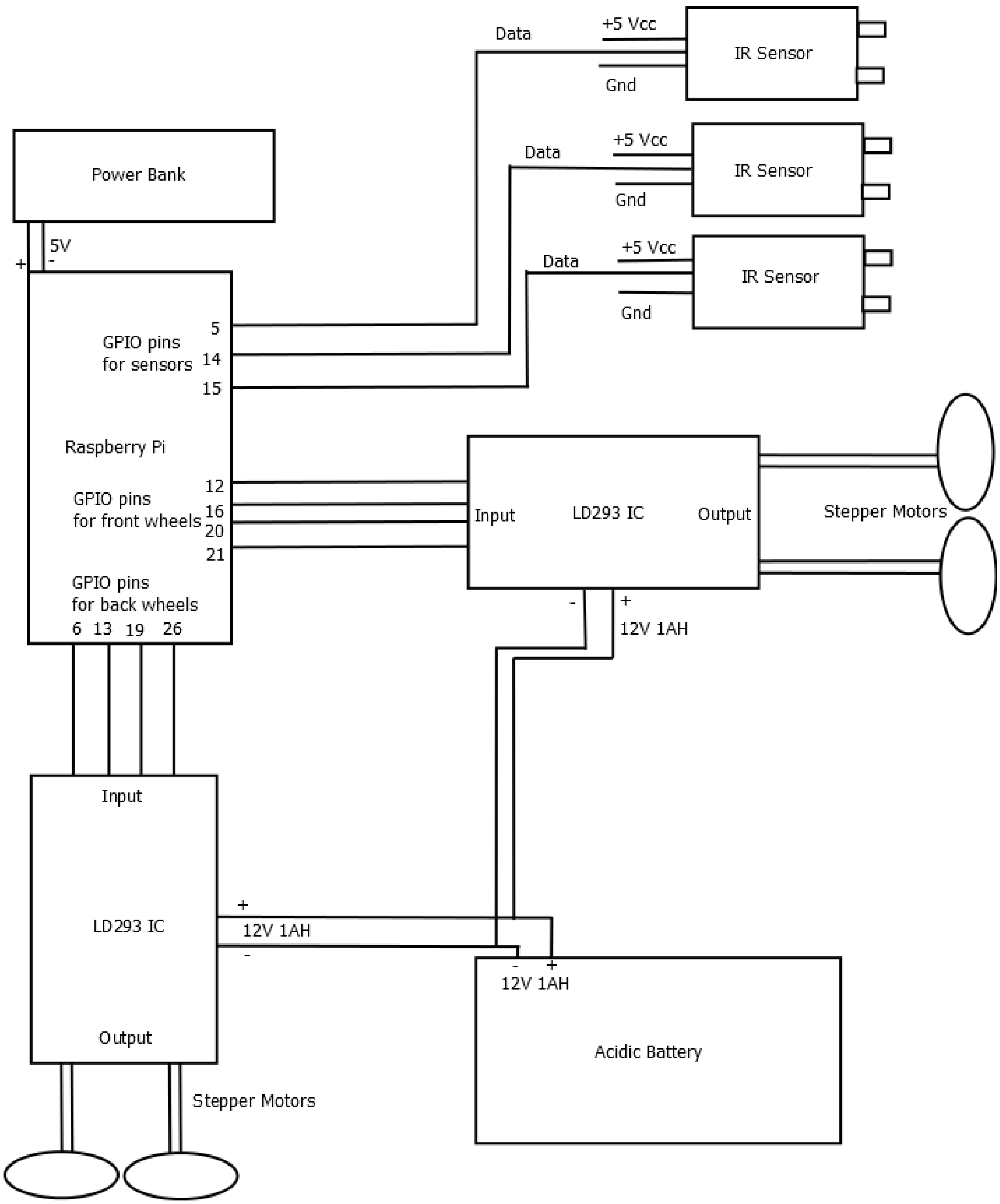

Fig.3 Circuit diagram

The Fig 1 shows the pictorial design of the system. Fig 2 shows the circuit diagram of the system. The system consists Raspberry Pi powered by power bank. The GPIO pins of the Raspberry Pi are used to get the input from the Photoelectric IR sensors which are in turn used for sensing the path, the GPIO pins are also used to give output to the LD293 IC which control the stepper motors. The stepper motors are powered by the 12V 1AH Acidic Battery.

5.4 Modules Design

1) IR Sensor Module

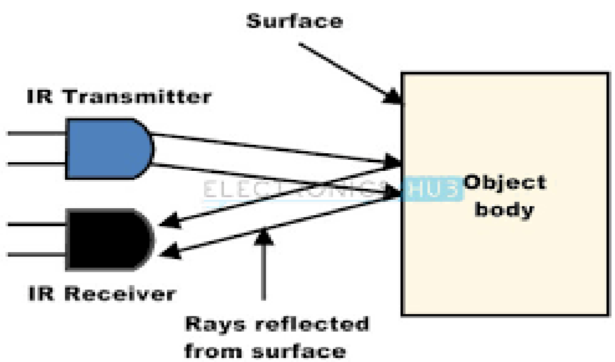

Fig.4 IR sensor working

Table No.3 Sensor truth table

Sensor	**Object**
Object	0
No Object	1

In the design of the path, the black tape will absorb the IR rays so the output will be 1 and the floor will reflect the IR rays so the output will be 0.The device consists of 3 Infrared Photoelectric Sensors. The middle sensor is on the black tape, left and right are not on the tape then the output of sensors will be (0 ,1, 0). And vehicle will travel in straight direction.If the left sensor is on the black tape then output will be (1, 0, 0) and device will turn left. Similarly, if right sensor on the black tape then the device will turn right (0, 0, 1).If all the sensors will come on the black tape then vehicle stop and wait for specific time for patient to pick up the medicine.

1) LD293IC and stepper motor module

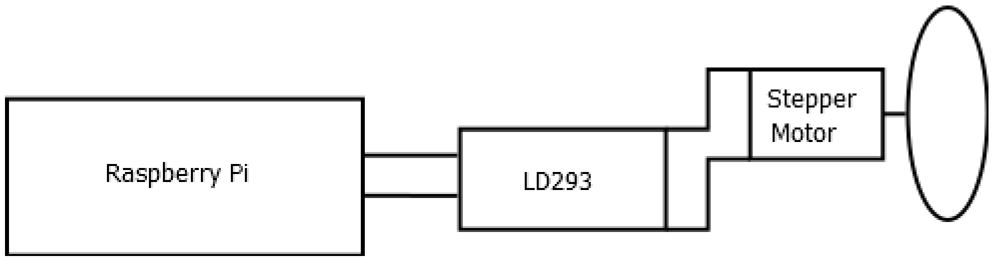

Fig.5 Block diagram of LD293 and stepper motor module

Table No.4 LD293 truth table

GPIO X	GPIO Y	Output
0	0	No Rotation
0	1	Clockwise Rotation
1	0	Anti-clockwise Rotation
1	1	No Rotation

The Fig 3 shows the pictorial design of the connection between the Raspberry Pi, LD293 IC and stepper motors. The LD293 IC requires two GPIO pins for each stepper motor to run them in required direction. The above table shows that the stepper motors will rotate only for unlike inputs and stop for like inputs. For moving straight input must be (Left wheels (1,0), Right wheels (1,0)), for left turn (Left wheels (1,1), Right wheels (1,0)), for right turn (Left wheels (1,0), Right wheels (1,1)), and to stop (Left wheels (1,1), Right wheels (1,1)).

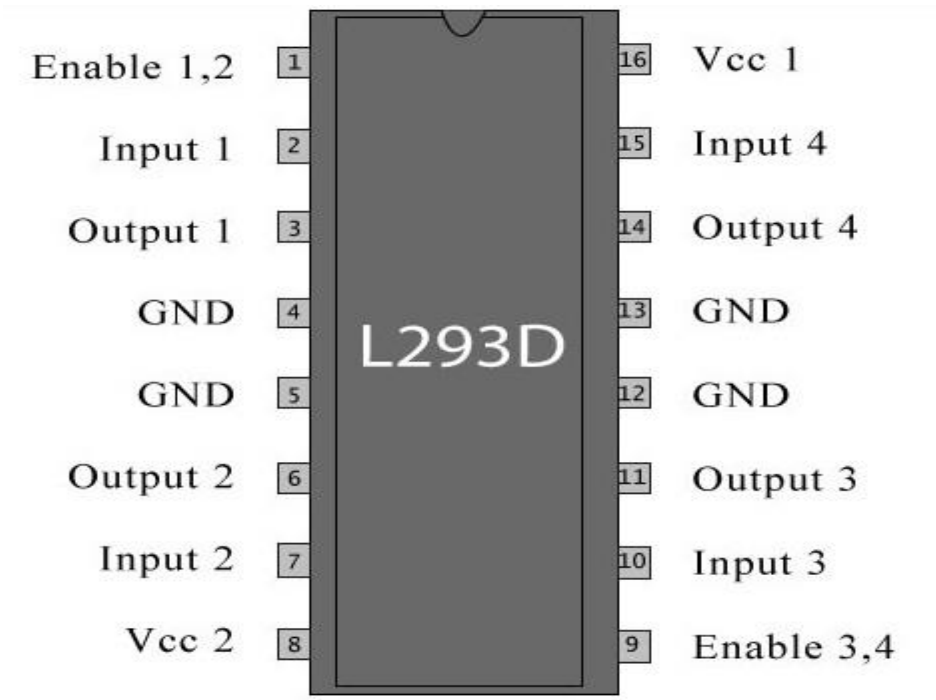

Fig.6 LD293 PIN diagram

A motor driver IC is an integrated circuit chip which is usually used to control motors in autonomous robots. Motor driver ICs act as an interface between microprocessors in robots and the motors in the robot.The L293D is a 16 pin IC, with eight pins, on each side, dedicated to the controlling of a motor. There are 2 INPUT pins, 2 OUTPUT pins and 1 ENABLE pin for each motor. L293D consist of two H-bridge. H-bridge is the simplest circuit for controlling a low current rated motor. The below table shows the pin characteristics of LD293 IC.

Table No.5 LD293 Pin characteristics

Pin No	Pin Characteristics
1	Enable 1-2, when this is HIGH the left part of the IC will work and when it is low the left part won’t work. So, this is the Master Control pin for the left part of IC
2	INPUT 1, when this pin is HIGH the current will flow though output 1
3	OUTPUT 1, this pin should be connected to one of the terminals of motor
4, 5	GND, ground pins

6	OUTPUT 2, this pin should be connected to one of the terminals of motor
7	INPUT 2, when this pin is HIGH the current will flow though output 2
8	VC, this is the voltage which will be supplied to the motor. So, if you are driving 12 V DC motors then make sure that this pin is supplied with 12 V
16	VSS, this is the power source to the IC. So, this pin should be supplied with 5 V
15	INPUT 4, when this pin is HIGH the current will flow though output 4
14	OUTPUT 4, this pin should be connected to one of the terminals of motor
13, 12	GND, ground pins
11	OUTPUT 3, this pin should be connected to one of the terminals of motor
10	INPUT 3, when this pin is HIGH the current will flow though output 3
9	Enable 3-4, when this is HIGH the right part of the IC will work and when it is low the right part won't work. So, this is the Master Control pin for the right part of IC

5.5 Conclusion

This chapter explains the system design and the two main modules which are IR Sensor Module and LD293 IC and stepper motor module that are used in the system.

Chapter 6

System Implementation

6.1 Introduction

This chapter explains the procedure of how to connect and test the IR sensors, Connect and test the LD293 IC and stepper motors and algorithmic procedure of line follower.

6.2 Connect and test the IR Sensors

Each IR sensor has three pins: VCC for power, GND for ground, and OUT for digital output. Connect the VCC to the 5V DC supply i.e. either to pin 2 or 4. Connect the GND to one of the ground pins (6, 9, 14, 20, 25, 30, 34, 39). The OUT should be connected to one of GPIO pins from which the input can be read. Then the sensors are tested by hovering them over the black tape. One of the light must turn off when the sensor is hovering on the black tape and both the lights should be on when it is hovering on some object.

6.3 Connect and test the LD293 IC and stepper motors

Single LD293 IC can control two stepper motors at once. The IC consists of four inputs and four outputs, two for each stepper motor. The inputs must be connected to the GPIO pins which will control in which direction the motor will rotate. The outputs of the IC must be connected to the stepper motors. The connections can be tested by sending (1, 0) as input, if the motor rotates in clockwise direction then for input (0, 1) it should rotate in anti-clockwise direction and vice versa. The motor should not rotate for inputs (1, 1) and (0, 0).

6.4 Program the line following algorithm

The algorithm reads the input from the IR Sensor and commands the LD293 IC how the wheels should rotate. For input (0, 1, 0) (left sensor, middle sensor, right sensor) all the wheels move forward, for input (1, N/A, 0) the left side wheels must stop making it to turn left, similarly for input (0, N/A, 1) the right side wheels must stop making it to turn right and for input (1, 1, 1) it should stop and wait for some time for the patients to pick up medicines.

Line Follower Algorithm

Input: Left and Right IR Sensors

Output: Wheel Movement

Method:

```
while True
    if leftSensor = 0 and midSensor = 1 and rightSensor = 0
        thenall four wheels will move forward
    else if leftSensor = 1 and rightSensor = 0
        thenleft side front and back wheels will stop, making it to turn Left
    else if leftSensor = 0 and rightSensor = 1
        thenright side front and back wheels will stop,making it to turn Right
    else if leftSensor = 1 and midSensor = 1 and rightSensor = 1
        thenwait and delivery the medicine
```

Results

The system follows the path and delivers the medicines to the patients on time and alerts them with an audio message to collect the medicine. The robotic system was designed to run for bus like topology i.e. the patient’s rooms are arranged according to bus topology as shown in the figure.

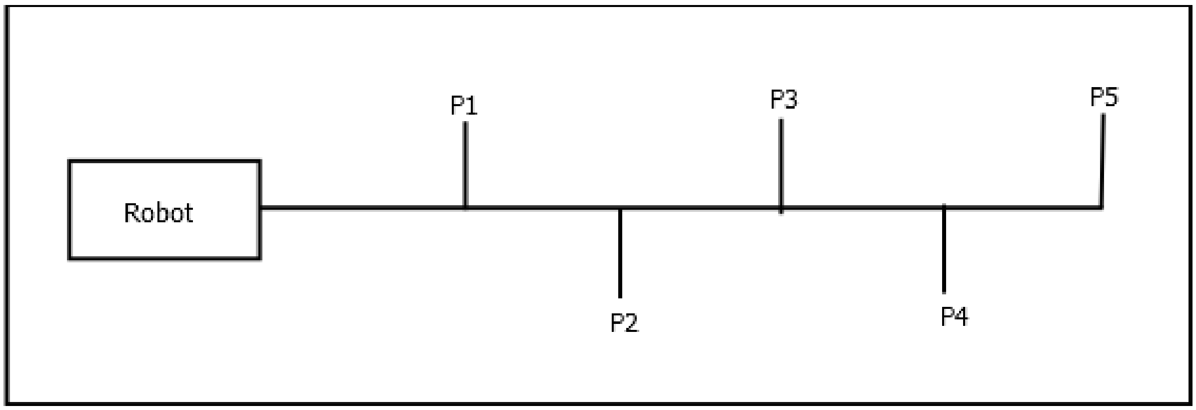

Fig.7 Room arrangement topology

Table No.6 Result analysis

Rounds	Number of medicine delivery stations	Expected number of stops	Observed stops	Accuracy
1	1(P1/P2/P3/P4/P5)	5	5	100%
2	2(P1,P2)/(P2,P3)/(P3,P4)/(P4,P5)/ (P2,P3)/(P2/P5)/(P1,P5)	7	6	85.7%
3	3(P1,P2,P3)/(P2,P3,P4)/(P3, P4, P5)/(P2,P4,P5)/(P1,P4,P5)	5	4	80%
4	4(P1,P2,P3,P4)/(P2,P3,P4,P5)/(P1,P3,P4,P5)	3	3	100%
5	5(P1,P2,P3,P4,P5)	1	1	100%

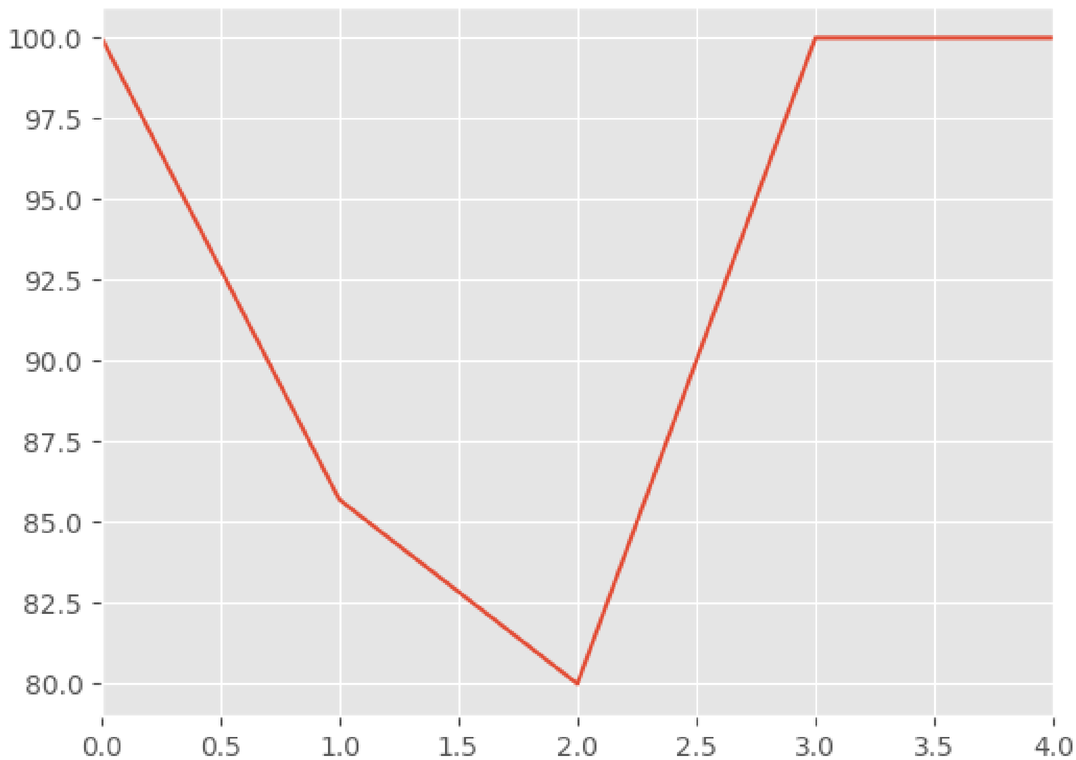

Fig.8 Accuracy graph

From the above experimental analysis, we can say that for number of patients equal to one we have 100% accuracy, for two we have 85.7%, for three we have 80%, for four we have 100% and for five we have 100% accuracy. So, on average we have at least 80% accuracy.

Conclusion and Future work

SmartMED can play a vital role in the field of hospitality. Robotics is a grooming technology. It can be very beneficially for the patients. Monitoring of every patient is very difficult for nurses in the hospital. So, a camera can be placed in the line following system, from which the status for every patient can be handled from a single room. The system can be modified to identify whether the patients collect their medicine or not and inform the doctors or nurse. From the above experimental analysis, we can conclude that our system works accurately for about 80% of time.

Snapshot

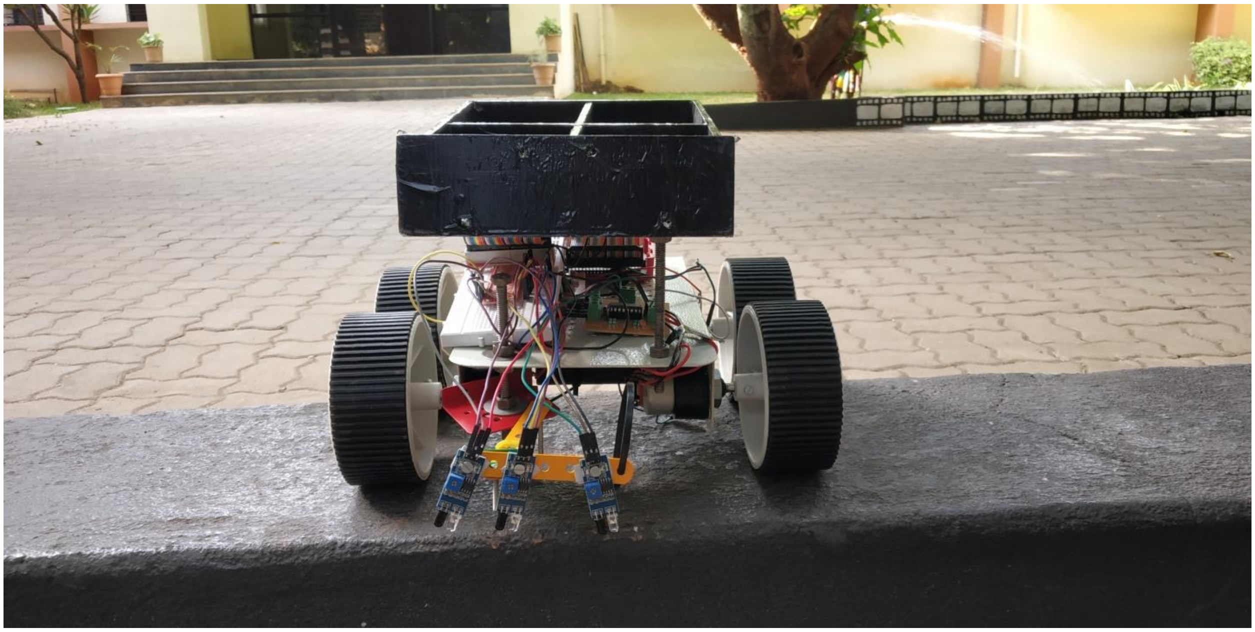

Fig. 9 Front view

Fig. 10 Back view

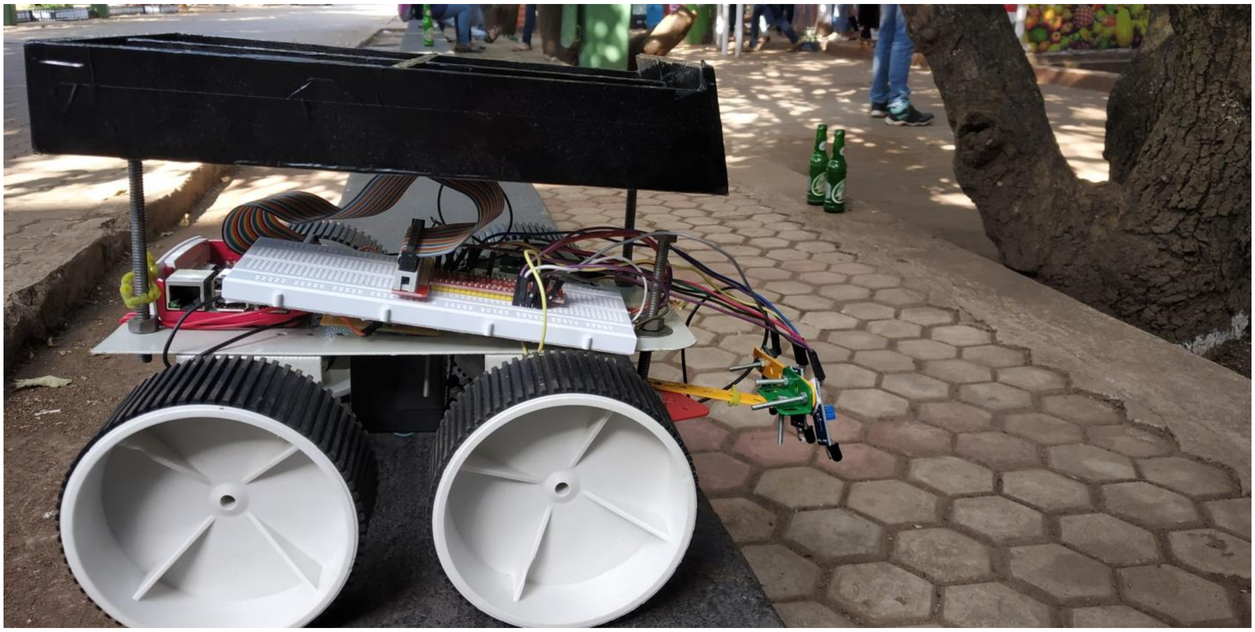

Fig. 11 side view

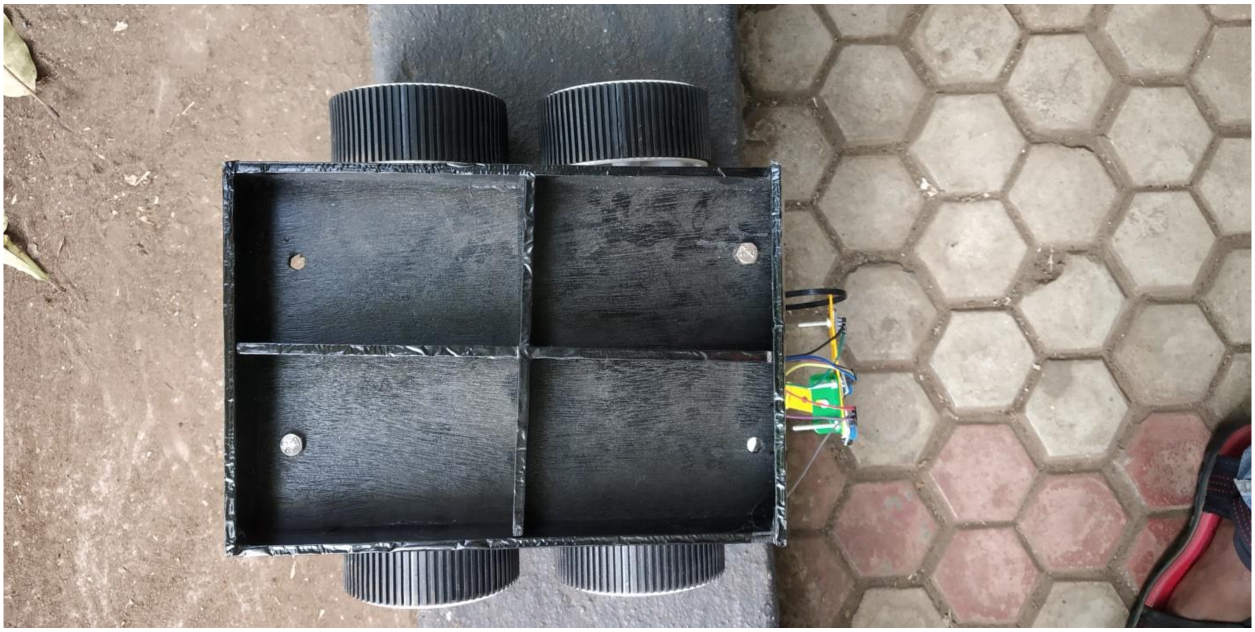

Fig. 12 Top view

Bibliography

[1] Fatima R. Ali, Abdulmuttalib T. Rashid "Design and implementation of static and dynamic objects store systems using line follower robots" https://ieeexplore.ieee.org/document/8370952.

[2] Md. Kamruzzaman, Md. Tareq, "Design and implementation of a robotic technique based waiter" https://ieeexplore.ieee.org/document/8275250.

[3] Wolfram Donat "Learn Raspberry Pi Programming with Python: Learn to Program on the World's Most Popular Tiny Computer" 2nd Edition.

[4] Bajestani, S.E.M., Vosoughinia, A., "Technical Report of Building a Line Follower Robot" International Conference on Electronics and Information Engineering, 2010

[5] Python documentation https://docs.python.org/3.

[6] PuTTY is an SSH and telnet client https://www.putty.org.

[7] Arshdeep Bahga and Vijay K. Madisetti, "Internet of Things: A Hands-on Approach", First edition (2015).

[8] David Hanes, Gonzalo Salgueiro, and Patrick Grossetete, "IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", 2017.

[9] Román Osorio," Inteligent Line Follower Mini-Robot System", International Journal of Computers, Communications & Control Vol. I, 2006.

[10] Raspberry Pi documentation https://www.raspberrypi.org.