TELE-PRESENCE TELE-ROBOT SYSTEM

Project report submitted in partial fulfillment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

IN

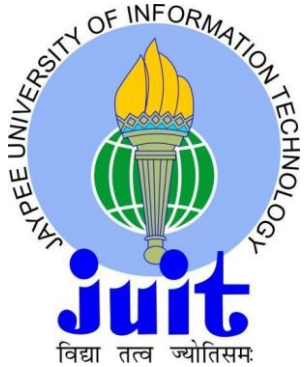
ELECTRONICS AND COMMUNICATION ENGINEERING

By

NISHANT AMBIYA (141004)
PRASHANT SHARMA (141088)

UNDER THE GUIDANCE OF

Mr. PARDEEP GARG



JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT
Month-May, Year-2018

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY

(Established by H.P. State Legislative vide Act No. 14 of 2002)
P.O. Waknaghat, Teh. Kandaghat, Distt. Solan - 173234 (H.P.) INDIA
Website: www.juit.ac.in
Phone No. (91) 01792-257999
Fax: +91-01792-245362

CERTIFICATE

This is to certify that the work reported in the B.Tech project report entitled **"TELE PRSENCE TELE ROBOTIC SYSTEM**" which is being submitted by Nishant Ambiya (141004), Prashant Sharma(141088) in fulfillment for the award of Bachelor of Technology in Electronics and Communication Engineering by the Jaypee University of Information Technology, is the record of candidate's own work carried out by him/her under my supervision. This work is original and has not been submitted partially or fully anywhere else for any other degree or diploma.

Mr. Pardeep Garg
Assistant Professor (Grade-II)
Department of Electronics & Communication Engineering
Jaypee University of Information Technology, Waknaghat

TABLE OF CONTENTS

DECLARATION BY THE SCHOLARS

We hereby declare that the work reported in the B-Tech thesis entitled **"TELE-PRESENCE TELE-ROBOT SYSTEM"** submitted at **Jaypee University of Information Technology, Waknaghat** is an authentic record of our work carried out under the supervision of **Mr. Pardeep Garg**. We have not submitted this work elsewhere for any other degree or diploma.

Nishant Ambiya

Prashant Sharma

Department of Electronics and Communication Engineering

Jaypee University of Information Technology, Waknaghat.

Date:

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the completion of the project would be incomplete without the mention of the people who made it possible.

We would like to take the opportunity to thank and express our deep sense of gratitude to our faculty mentor **Mr. Pardeep Garg** for providing his valuable guidance at all stages of the study, his advice, constructive suggestions, positive and supportive attitude and continuous encouragement, without which it would have not been possible to complete the project.

We are obliged to all the faculty members of JUIT, for the valuable information provided by them in their respective fields. We are grateful for their cooperation during the period of their project. We hope that they can build upon the experience and knowledge that they have gained and make a remarkable mark in the future.

LIST OF ACRONYMS AND ABBREVIATIONS

WebRTC	Web Real-Time Communication
DC	Direct Current
EN	Enable Pin
GPU	Graphic Processing Unit
BCM	Body Control Module
RAM	Random Access Memory
GB	Giga Bytes
LAN	Local Area Network
BLE	Bluetooth Low Energy
USB	Universal Serial Bus
HDMI	High-Definition Multimedia Interface
CSI	Camera Serial Interface
DSI	Display Serial Interface
SD	Secure Digital
12C	Inter-IC
SPI	Serial Peripheral Interface
LED	Light Emitting Diode
IP	Internet Protocol
UI	User Interface
WLAN	Wireless Local Area Network
GPIO	General-purpose input/output

LIST OF FIGURES

ABSTRACT

Telepresence is a technology in which the person operating it feels or has a sense of being present in that location Telepresence system helps the user to have a presence in remote locations and to interact with other people between long distance. A user of this system only needs to know how to drive this system in order to be in the remote environment. This experience also resembles virtual reality. The remotely controlled robot is also called tele robot and the human operator can be located at a very far away distance. The control and feedback mechanism is done over wireless network.

Many people think telepresence and video conferencing is the same thing. A video conference is a type of communication which includes communication using image and audio. However, telepresence is basically being in a remote location and even though it is not completely related to video conferencing, it also includes robotic telepresence.

CHAPTER 1
INTRODUCTION

1.1 Introduction

Telepresence tele-robot system can be understood by breaking up the words i.e. telepresence and tele-robot [1].

Telepresence has been defined as the human experience of being present at a live location away from location where he is being present. Someone using this telepresence system would be able to act and receive stimuli in similar way as though he is present in that remote location. And hence it would help users to be part in interactive group activities from a remote location and thus this system has a wide range of applications [2].

Tele-robotics is defined as the area of robotics that is concerned with the control of the autonomous robots from a distant location. It is mainly controlled using the wireless communication networks like WLAN, Bluetooth, Wi-Fi, cellphone network and similar other things) or using wired connections. It is basically the composition of the subfields, teleoperation and telepresence [3].

This telepresence tele robot system is a fully robotic platform that pairs with your Android phone or tablet (running our free App) to become a complete telepresence solution. It connects remotely to the robot system from anywhere in the world on most any device with an Internet connection (Android, iOS, Windows, Mac). Since it uses WebRTC technology, the android interface is simple and light (doesn't require any additional plugins or java applets). It is a fast and responsive system with minimal lag over broadband connections. It will include software made in android interface which is used to run the robot and will be a two-way audio and video communication [4].

1.2 Motivation

Telepresence is a wonderful technology which is becoming essential nowadays. It has many benefits which have been discussed in objectives. Not just in businesses, but in all aspects of our life telepresence can be used. However, the basic infrastructure required for telepresence is high bandwidth connection, speakers, microphones, high definition camera and a few other tools. With these in place, we can make communication to people is remote areas without interruption. The usage of telepresence is increasing and with gradual increase in internet speed we should see it becoming essential in few years' time.

1.3 Objectives

Main objectives of telepresence tele-robot system are [5]:

- **Full-Fill Business Objective**

It helps in communication between two people. For business it leads to smooth communication between employees and employers. You can share your ideas, files, videos and other things on the spot. It saves time and is ultimately value for the business.

- **Less Expenditure**

A telepresence meeting has a conference room in which the participants get together. It is like being in a place where you are not physically present. This will save your expenditure. You do not have to travel to different destination for meetings. It saves a lot of travelling and accommodation cost. It can be used to cut down overhead expenditure.

- **Globalization**

Today everything we do in the world and among countries are a part of globalization and globe equality. Imagine a doctor being able to help another person in an under developed nation, researchers helping out people in other countries, people sharing their knowledge and experiences etc. This is the reason why there has been a boost in telepresence rooms in developed countries like US and UK. The governments of many countries have also started using this technology for communication between different ministries

.

- **Increase in Productivity**

In an organization, telepresence will help I better coordination between clients, employees and suppliers. It can boost productivity by up to 40 per cent due to quick decision making, saving travel cost, saving travel time.

- **Recruitment Made Easy**

The method of employing employees is a very lengthy procedure. It takes time and is limited to only areas of physical reach. You can schedule multiple interviews with people and can also save the video for later reference. This can help a business to evaluate the best candidate. You can also recruit from the world and choose among many people.

1.4 System Layout

The Tele presence robot consist of the Raspberry pi 3B Board interfaced with the L293D motor driver integrated chip, which is connected to DC motors. The display and camera are used by Android phone which are connected to Raspberry pi through WLAN. A user interface (UI) on an Android enabled mobile offer connection with the system robot and provide control operations. The camera and display offers the two-way communication both audio and video.

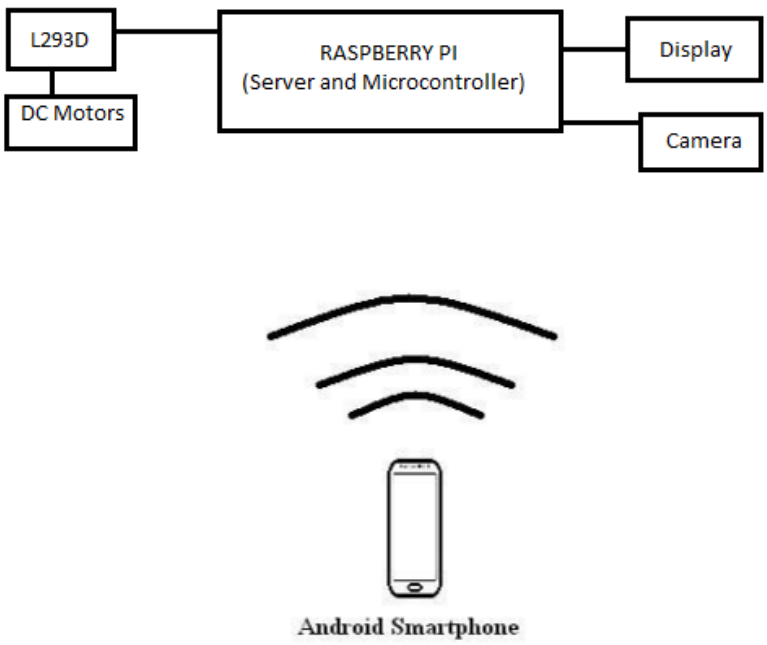


Figure 1.1: System layout

1.5 Literature Review

In this paper [6] the design is implemented that controls and monitors the tele robot system using the web based application. It enhances the user experience by making the controller easy to use, easy to handle, portable, easy to access and hence makes it appropriate to use. All the previous used tele robotic control system were desktop or PC based which were installed in desktop and hence were not portable. This system is made using web software development. It just uses the touch of the figure to direct the robot in particular direction. Its controls are performed by "Tech bot Web User Interface" which is implemented using JavaScript and PHP. The web based interface does not need prior installation as it can be accessed from anywhere just using the web browser. Hence this this system can be accessed from desktop or smart phone just with the help of web browser with internet connection.

This research [7] focuses on how a telepresence robotic system, the user which controls the robot and the people around the robot collaborate with each other so that robot reaches its desired location. This system follows the person by autonomous navigation. It contains the sensors that do processing that accurately tell obstacles. This research is focused on two interactive design of telepresence robot "Hugo" and "Margo" of VGo robots. There are many commercial robots in the market which are just made for masses like doctors, caregivers, corporates etc [8]. Most of these only have arrow keys that will move the robot. It has live video conferencing that user see in remote location and navigate with it. And due to advancement in mobile technology the robot can be navigated from anywhere as it just needs good internet connection with available bandwidth and low packet loss. Some robots are there in market which provides assisted navigation which helps to move

them through obstacles and balance in case of varied terrain. But doing all this robot is still not able to move safely or accepted by others socially. Hence this research mainly focuses on navigating the environment in safe and socially manner using both techniques automatic and semi-automatic. Its appearance must be given as importance as it is given to its functions and technology.

CHAPTER 2

HARDWARE SPECIFICATIONS

2.1 L293D

To drive the DC motors, we require L293D devices in our framework. These are used as the motor drivers, which are quadruple high current half-H drivers. These motor drivers take low power signals as input and convert them into high power signals as output. These drivers act as amplifier. The L293D is designed in such a way that they can provide bidirectional current up to the range of 600mA with voltage range from 4.5V to 36V. The two H-bridge circuits inside this chip helps to drive the motors synchronously in both the direction. There are four drivers in each L293D chip. The driver 1 and driver 2 are enabled by 1,2EN pins while driver 3 and driver 4 are enabled by 3,4EN pins. The input control pins are 2, 7, 10, 15 while their output pins are 3, 6, 11, 14 respectively [9].

The DC motor cannot be interfaced directly with the GPIO of Raspberry pi as GPIO only gives maximum current up to 16mA at voltage 3.3V. Whereas the motors require high measure of current ranges from 60mA to 300mA at 12V. Hence the L293D motor driver is used to drive the DC motors.

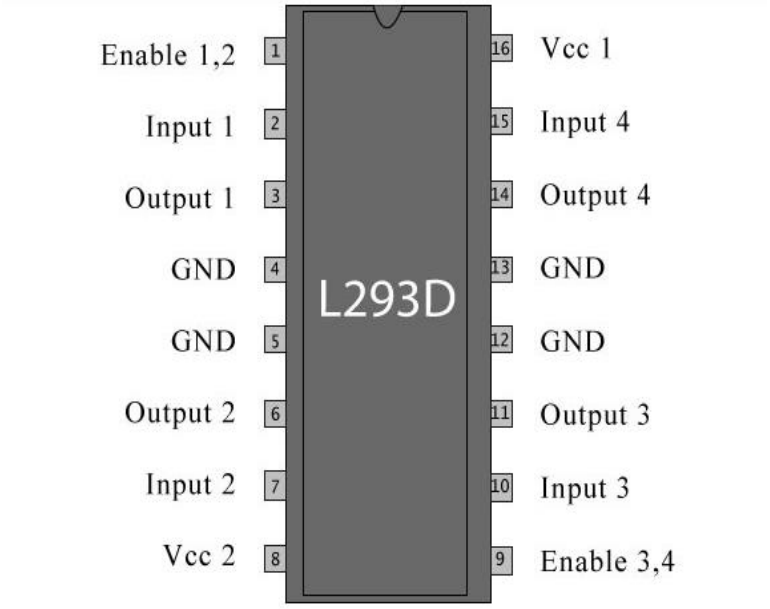


Figure 2.1: L293D

2.4 DC MOTORS

Used 6 DC motors

5 100RPM 12V DC Motor and 1 L-shaped DC Motor

2.4.1 12V DC motors

- Weight 125 grams.
- shaft diameter of 6mm
- speed of 100rpm
- torque of 1.2 kg-cm
- No load current of 60mA max and load current of 300mA max.

DC motors are required for the mobility of the telepresence robot. With the help of motors, the robot is able to move forward, backward and even turns left or right [10].

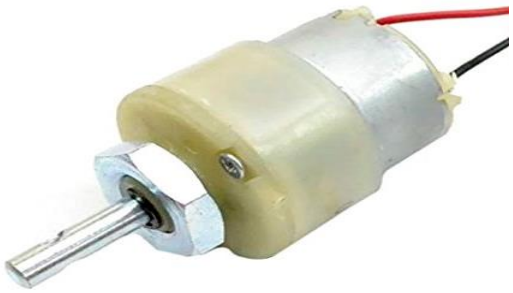


Figure 2.2: 12 V DC Motor

2.4.2 L-Shaped DC Motor

The L-shaped motor runs on a nominal voltage of 5V and nominal current upto mA. The shaft diameter is of 5.5mm. The speed of motor is 180 rpm [10].

Figure 2.3: L-shaped DC Motor

This motor is used to handle the movements of the upper part of telepresence robot. It is used in the neck of robot which is used to move telepresence robot's head in upward or downward direction. The neck contains the support for screen. This neck is mounted on a stick. The upper edge of stick contains the screen while the lower edge has a DC motor attached to it. This DC motor moves the neck in direction left and right. All this

functionality which is implemented will give robot mobility like human neck i.e. rotating the screen in all the three directions.

2.6 Raspberry pi 3 model B

The Raspberry Pi 3 Model B is the earliest model of the third-generation Raspberry Pi board. To interface the telepresence robot hardware and software together it would need a microcontroller. This is implemented using Raspberry pi. Not only it is used as microcontroller but also as a server. It holds all the code i.e. software that are used to control the mobility of telepresence robot. The Raspberry pi contains a Raspbian OS which is installed in the SD card. This SD card contains all the software of the telepresence robot [11].

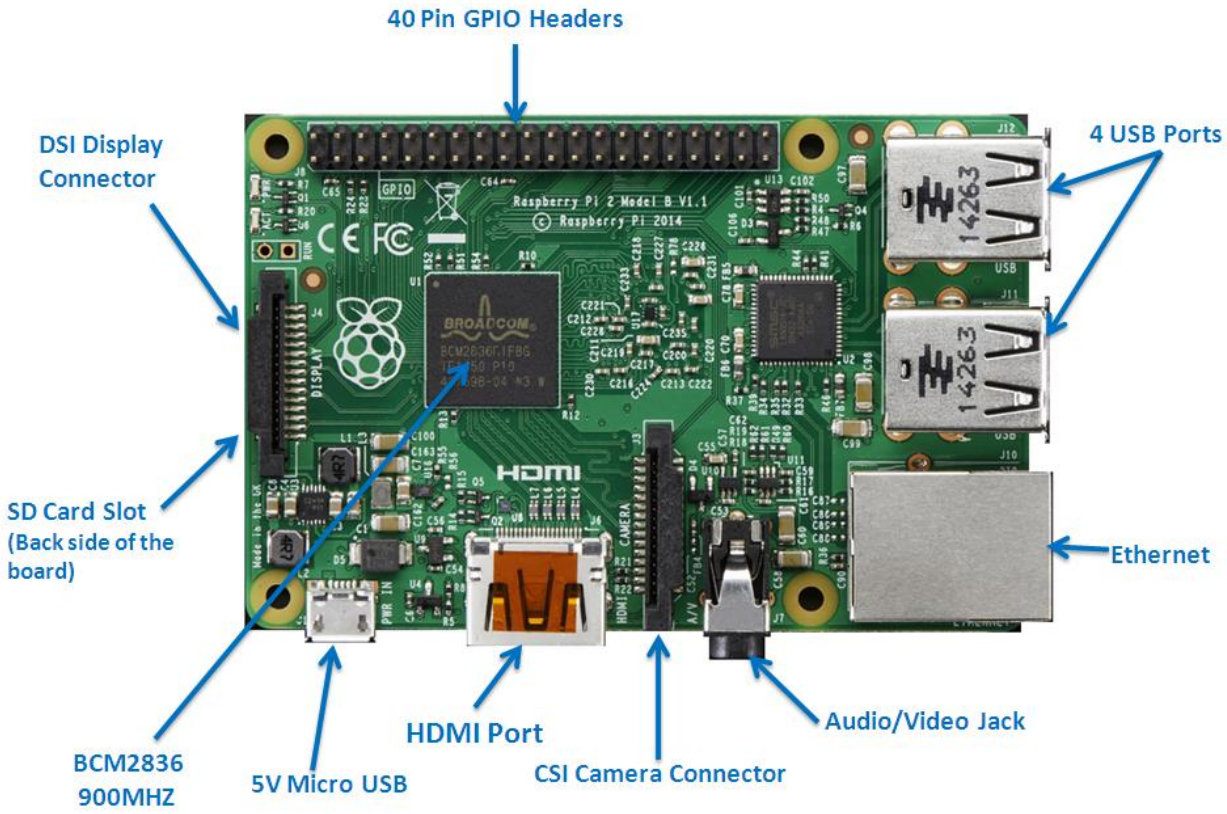


Figure 2.3: Raspberry pi 3B

Operating System:

Operating system installed in SD card is a Debian-based Linux operating system.

GPIO pins:

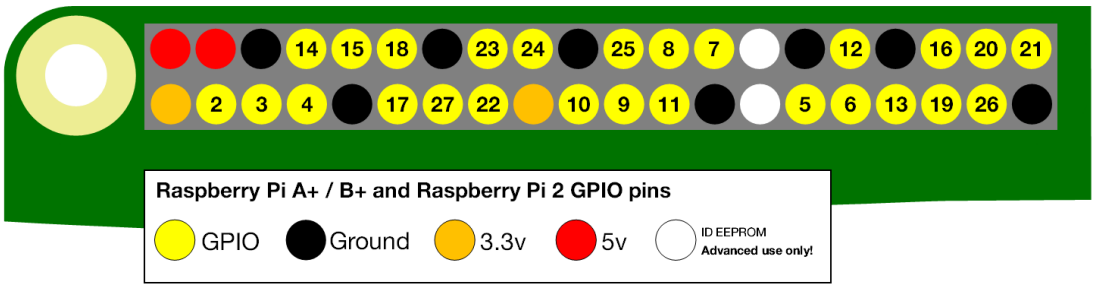


Figure 2.5: GPIO Pins

The most important feature of raspberry pi is GPIO (general Purpose Input/Output) pins. Any of the GPIO pins can be assigned (in programming) as an information or yield stick and utilized for an extensive variety of purposes. The numbering of the GPIO pins isn't in numerical order; GPIO pins 0 and 1 are available on the board (physical pins 27 and 28) however are saved for other functions [11].

Output:

The high level output is 3.3V whereas low level output is 0V.

Input:

A GPIO pin can read input of voltage 3.3V and 0V.

The output of GPIO pins are used to drive the motor driver of the telepresence robot.

2.4 HARDWARE INTERFACING

The circuit diagram for the hardware is given below:

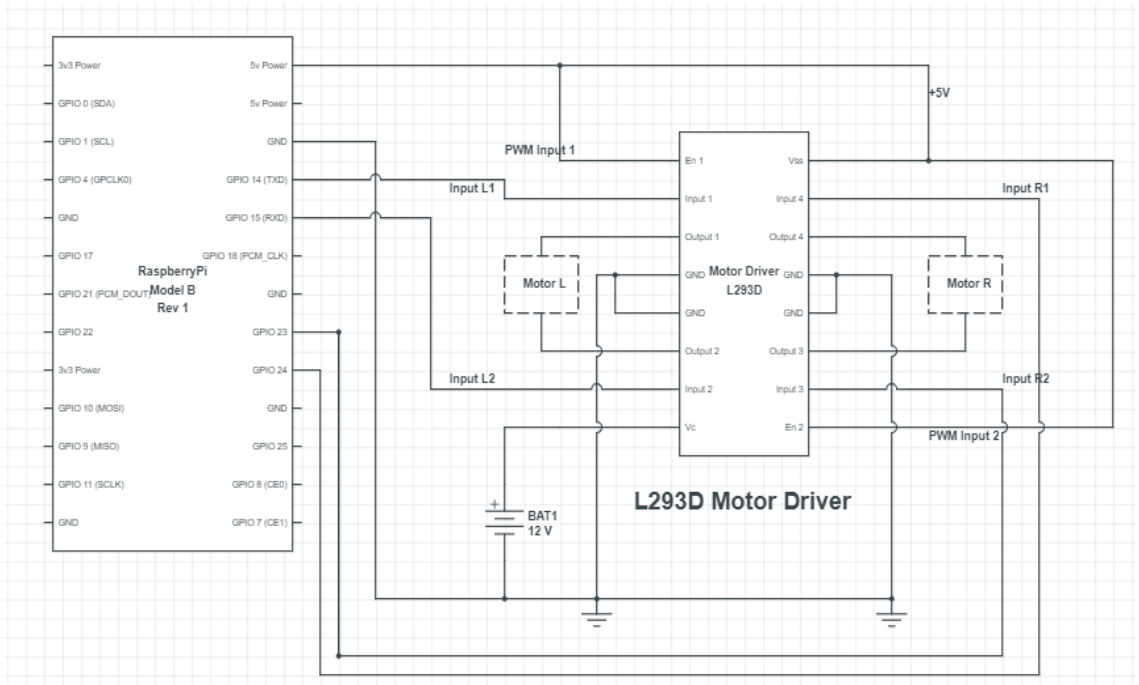


Figure 2.6: Circuit layout

The components used in this hardware circuit are raspberry pi 3 board, L293D motor driver, battery for power supply, DC motors and connecting wires for the connections. The raspberry pi act as microcontroller that generates controlling signals of 3.3V through GPIO pins. Motor driver which is L293D takes the signal from raspberry pi as input and output the high power signal delivered by raspberry pi to the DC motors. The raspberry pi gets input power of 5V whereas the motor driver (L293D) gets power V_c 12V and V_{ss} 5V from battery.

CHAPTER 3

SOFTWARE SPECIFICATIONS

3.1 Pi4J Java library

Pi4J is an open source java library that is used to program the GPIO pins of raspberry pi. The program for controlling the GPIO pins is made using this library. To communicate with the android application, java socket programming is being implemented. It uses internally WiringPi library that is used to access the GPIO pins of it. This library is written in C for all versions of Raspberry Pi. The read all command gives the description of all the pins of the Raspberry pi [12].

Tele.java program is implemented which is used to configure the GPIO pins of raspberry pi [13].

```
// Server Program for Tele Robot
import java.net.*;
import java.io.*;

public class GetCommand {

   //Initialize socket and input output streams
   private Socket socket = null;
   private ServerSocket server = null;
   private DataInputStream recieveFromSocket = null;
   private DataOutputStream sendToSocket = null;
   private Tele t = new Tele();
   private String ReplyToClient;

   public void StartServer(int port)
   {
      try{
         server = new ServerSocket(port);

         System.out.println("Server Started");

         while(true){
         socket = server.accept();

         System.out.println("Client Connected");
```

```
            recieveFromSocket = new DataInputStream(new BufferedInputStream
            (socket.getInputStream()));

           String MessageFromClient = recieveFromSocket.readUTF();

           switch(MessageFromClient){
              case "forward":
                 {t.forward();ReplyToClient="success"; break;  }
              case "backward":
                 {t.backward();ReplyToClient="success";break;  }
              case "stop":
                 {t.stop();ReplyToClient="success";break;}
              case "right":
                 {t.right();Thread.sleep(200);t.stop();ReplyToClient="success";break;}
              case "left":
                 {t.left();Thread.sleep(200);t.stop();ReplyToClient="success";break;}
              case "exit":
                 { System.out.println("Server Exiting");System.exit(0);}
              default:{ReplyToClient="fail";}
           }

           sendToSocket = new DataOutputStream(socket.getOutputStream());
           sendToSocket.writeUTF(ReplyToClient);

           socket.close();
           recieveFromSocket.close();
           sendToSocket.close();
           }
        }
        catch(Exception e)
        {
           //
           System.out.println("Exception : "+e.getMessage());
        }
     }

     public static void main(String[] args){
      GetCommand g = new GetCommand();
      g.StartServer(12345);
     }
}
```

The program “Tele.java” uses Pi4J library for running the motors

```
import com.pi4j.io.gpio.*;
import java.util.Scanner;

/*

GPIO 0 = Physical 11
GPIO 1 = Physical 12 (PWM0)
GPIO 2 = Physical 13
GPIO 3 = Physical 15
```

```
GPIO 4 = Physical 16
GPIO 5 = Physical 18
GPIO 6 = Physical 22
GPIO 21 = Physical 29
GPIO 22 = Physical 31
GPIO 23 = Physical 33 (PWM1)
GPIO 24 = Physical 35
GPIO 25 = Physical 37
GPIO 26 = Physical 32 (PWM0)
GPIO 27 = Physical 36
GPIO 28 = Physical 38
GPIO 29 = Physical 40

*/
public class Tele{

GpioController gpio;

GpioPinDigitalOutput pin1;
GpioPinDigitalOutput pin2;
GpioPinDigitalOutput pin3;
GpioPinDigitalOutput pin4;

        public Tele()
        {
                gpio = GpioFactory.getInstance();

                pin1 = gpio.provisionDigitalOutputPin(RaspiPin.GPIO_00,"Motor1-
Input1",PinState.LOW);
                pin2 = gpio.provisionDigitalOutputPin(RaspiPin.GPIO_01,"Motor1-
Input2",PinState.LOW);

                pin3 = gpio.provisionDigitalOutputPin(RaspiPin.GPIO_02,"Motor2-
Input1",PinState.LOW);
                pin4 = gpio.provisionDigitalOutputPin(RaspiPin.GPIO_03,"Motor2-
Input2",PinState.LOW);

                pin1.setShutdownOptions(true,PinState.LOW);
                pin2.setShutdownOptions(true,PinState.LOW);

                pin3.setShutdownOptions(true,PinState.LOW);
                pin4.setShutdownOptions(true,PinState.LOW);
        }

        boolean forward() throws Exception
        {
                pin2.low();
                pin4.low();
                pin1.high();
                pin3.high();

                return true;
        }
```

```
        boolean backward() throws Exception
        {
                pin1.low();
                pin3.low();
                pin2.high();
                pin4.high();

                return true;
        }
        boolean right() throws Exception
        {
                pin1.low();
                pin4.low();
                pin2.high();
                pin3.high();

                return true;
        }
        boolean left() throws Exception
        {
                pin2.low();
                pin3.low();
                pin1.high();
                pin4.high();

                return true;
        }
        boolean stop() throws Exception
{
                pin1.low();
                pin3.low();
                pin2.low();
                pin4.low();

                return true;
        }
        public static void main(String args[]) throws Exception{

                Tele t = new Tele();

                while(true){
                System.out.println("Choose Option");
                System.out.println("1. Forward");
                System.out.println("2. Backward");
                System.out.println("3. Stop");
                System.out.println("4. Exit");
                Scanner s = new Scanner(System.in);
                int c = s.nextInt();
                 switch(c){
                        case 1 : t.forward();
                                 break;
                        case 2 : t.backward();
                                 break;
                           case 3 : t.stop();
```

```
                    break;
                case 4 : System.exit(0);
                default : System.out.println("Incorrect Option");
            }
        }
    }
}
```

GPIO read all

```
+-----+-----+---------+------+---+---Pi 3---+---+------+---------+-----+-----+
| BCM | wPi |   Name  | Mode | V | Physical | V | Mode | Name    | wPi | BCM |
+-----+-----+---------+------+---+----++----+---+------+---------+-----+-----+
|     |     |    3.3v |      |   |  1 || 2  |   |      | 5v      |     |     |
|   2 |   8 |   SDA.1 |   IN | 1 |  3 || 4  |   |      | 5v      |     |     |
|   3 |   9 |   SCL.1 |   IN | 1 |  5 || 6  |   |      | 0v      |     |     |
|   4 |   7 | GPIO. 7 |   IN | 1 |  7 || 8  | 1 | ALT0 | TxD     | 15  | 14  |
|     |     |      0v |      |   |  9 || 10 | 1 | ALT0 | RxD     | 16  | 15  |
|  17 |   0 | GPIO. 0 |  OUT | 0 | 11 || 12 | 0 | IN   | GPIO. 1 | 1   | 18  |
|  27 |   2 | GPIO. 2 |   IN | 0 | 13 || 14 |   |      | 0v      |     |     |
|  22 |   3 | GPIO. 3 |   IN | 0 | 15 || 16 | 1 | OUT  | GPIO. 4 | 4   | 23  |
|     |     |    3.3v |      |   | 17 || 18 | 1 | OUT  | GPIO. 5 | 5   | 24  |
|  10 |  12 |    MOSI | ALT0 | 0 | 19 || 20 |   |      | 0v      |     |     |
|   9 |  13 |    MISO | ALT0 | 0 | 21 || 22 | 0 | IN   | GPIO. 6 | 6   | 25  |
|  11 |  14 |    SCLK | ALT0 | 0 | 23 || 24 | 1 | OUT  | CE0     | 10  | 8   |
|     |     |      0v |      |   | 25 || 26 | 1 | OUT  | CE1     | 11  | 7   |
|   0 |  30 |   SDA.0 |   IN | 1 | 27 || 28 | 1 | IN   | SCL.0   | 31  | 1   |
|   5 |  21 | GPIO.21 |   IN | 1 | 29 || 30 |   |      | 0v      |     |     |
|   6 |  22 | GPIO.22 |   IN | 1 | 31 || 32 | 0 | IN   | GPIO.26 | 26  | 12  |
|  13 |  23 | GPIO.23 |   IN | 0 | 33 || 34 |   |      | 0v      |     |     |
|  19 |  24 | GPIO.24 |   IN | 0 | 35 || 36 | 0 | IN   | GPIO.27 | 27  | 16  |
|  26 |  25 | GPIO.25 |   IN | 0 | 37 || 38 | 0 | IN   | GPIO.28 | 28  | 20  |
|     |     |      0v |      |   | 39 || 40 | 0 | IN   | GPIO.29 | 29  | 21  |
+-----+-----+---------+------+---+----++----+---+------+---------+-----+-----+
| BCM | wPi |   Name  | Mode | V | Physical | V | Mode | Name    | wPi | BCM |
+-----+-----+---------+------+---+---Pi 3---+---+------+---------+-----+-----+
```

Figure 3.1: GPIO Pin status

3.2 ANDROID APPLICATION

First activity:

[14] The first activity contains the IP address and port. The IP is IVP4 address of the telepresence robot and the port number of telepresence robot through which the information is exchanged between the android application and telepresence robot

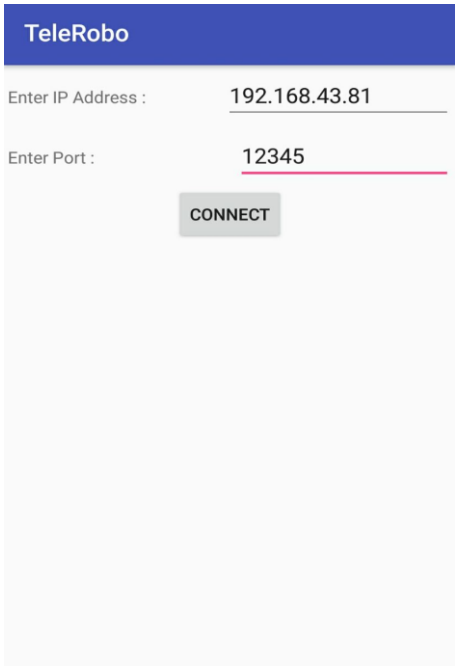


Figure 3.2: TeleRobo.MainActivity

Second activity:

The second activity contains the controllers that are used to run the robot in all the directions.

Figure 3.3: TeleRobo.Connect

CHAPTER 4

RESULTS AND DISCUSSIONS

The objective of this project is fulfilled by designing the telepresence robot. The user effectively controls and monitors the movements of telepresence robot using an android application. The control operation of telepresence robot depends upon the availability of the network. So it can be controlled from anytime, anywhere where network is available. This telepresence system has both live audio as well as video streaming features that makes this system livelier. The various possibilities are that in addition to the user directing the robot, many other users were able to log on and view the video as well. This makes this tele robotic system highly accessible to both users who want to direct the telepresence robot. During trial runs, there were instances when the screen mounted on system was unstable. So we used light weight screen.

Figure 4.1: Tele Robot

The design of our User Interface is based upon using an Android application, this user interface is better that the HTML version. Hence this frees the Android Application from having to send HTML web pages to browsers, allowing the telepresence robot Android Application and the robot to solely process messages sent by the user. The Telepresence robot Android Application provided an additional benefit to this system by allowing the telepresence robot to only maintain one session, regardless of the number of users. This makes this tele robotic system scalable. The GSM module can be applied in place of Wi-Fi module to enhance the range of telepresence connectivity.

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 CONCLUSION

We have presented the design of this robot as the standard semiautonomous telepresence robot. The things learnt during the implementation of this project will be implemented in the next design.

This system can be used in variety of domains such as navigation, human-robot interaction and recognition. We have briefly described the robot in our current work so as to achieve a user friendly semi-autonomous telepresence robotic system. The telepresence robot has a user friendly, easy to use control system. While some of the research challenges might already be solved but integration of these solutions in a single system might be a challenging task.

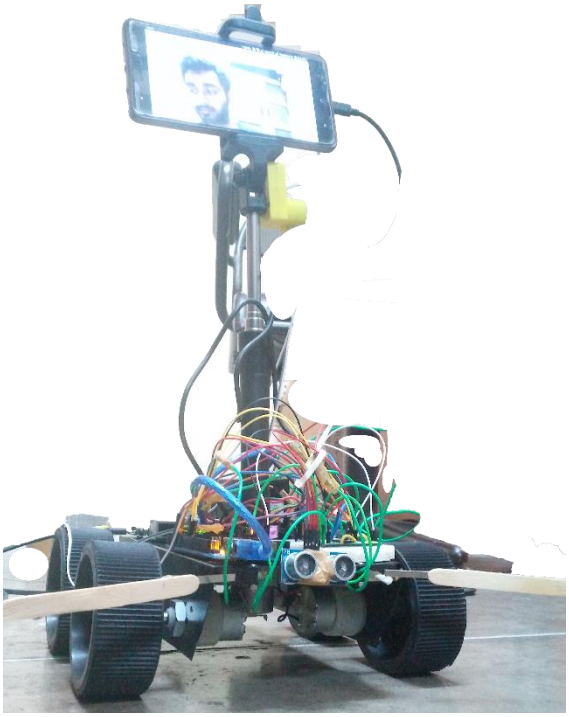


Figure 5.1: Tele Robot

5.2 FUTURE SCOPE

The things learnt during the implementation of this project will be used in the next design. We will be able to implement both autonomous and semi-autonomous system. It will support follow the person and reach the destination function automatically. This autonomous function can be designed using machine learning, artificial intelligence and neural networks. It will support collision avoidance and automatic driving. The physical structure of the robot can be improved for performing various types of tasks like picking items, opening door etc.

REFERENCES

[1] Illah R. Nourbakhsh "*Using robotic tele-embodiment to bring insects face to face with humans*" The Robotics Institute Carnegie Mellon University Pittsburgh, Pennsylvania

[2] Telepresence robot description "https://en.wikipedia.org/wiki/Telepresence"

[3] Telepresence difference "https://www.eztalks.com/"

[4] Dongjie Li, Sch. of Autom, Harbin Univ. of Sci. & Technol, Harbin, China Li Zhang ; Bo You ; Zhilong Yu, "*Research on telerobot control modes and timely automatic switching strategy*"

[5] Objective Telepresence robot "https://searchenterpriseai.techtarget.com"

[6] A. John Paul Antony, Dr. S. Rajpandian "*Web-Based Control and Monitoring of Telepresence Robot*"

[7] Katherine M. Tsui, Adam Norton, Daniel J. Brooks. Eric McCann,Mikhail S. Medvedev, Jordan Allspaw, Sompop Suksawat, James M. Dalphond, Michael Lunderville, Holly A. Yanco "*Iterative design of a semi-autonomous social telepresence robot research platform: a chronology*"

[8] Desai M, Tsui K, Yanco H, Uhlik C (2011)" *Essential features of telepresence robots. In*": Proceedings of IEEE Conference on technologies for practical robot applications (TePRA)

[9] L293, L293D Data Sheet by Texas Instruments "SLRS008D –SEPTEMBER 1986–REVISED JANUARY 2016"

[10] 12 V DC Motor "https://robokits.co.in"

[11] Raspberry pi hardware Documentation "www.raspberrypi.org/raspberrypi/"

[12] Pi4J library "pi4j.com"

[13] Java Socket Programming www.geeksforgeeks.com

[14] Android developer's guides "https://developer.android.com/guide/"

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT
LEARNING RESOURCE CENTER

PLAGIARISM VERIFICATION REPORT

Date: 18-05-18

Type of Document (Tick): Thesis | M.Tech Dissertation/ Report | B.Tech Project Report ✓ | Paper

Name: Nishant Ambiya , Prashant Sharma Department: ECE

Enrolment No. 141004 , 141088 Registration No. _____

Phone No. 9882051823 Email ID. nishantambiya@gmail.com

Name of the Supervisor: Pardeep Garg

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): Telepresence Tele Robotic System

Kindly allow me to avail Turnitin software report for the document mentioned above.

Nishant Prashant
(Signature of Student/s)

18/5/18
(Signature of Guide/Supervisor)

FOR ACCOUNTS DEPARTMENT:

Amount deposited: Rs. 300/- Dated: 18/5/2018 BRN1805514.
(Enclosed payment slip)

(Account Officer)

FOR LRC USE:

The above document was scanned for plagiarism check. The outcome of the same is reported below:

Copy Received on	Report delivered on	Similarity Index in %	Submission Details	
19/05/2018	21/05/2018	8%	Word Counts	3,929
			Character Counts	20,786
			Page counts	34
			File Size	1.06M

Checked by
Name & Signature

Librarian

LIBRARIAN
LEARNING RESOURCE CENTER
Jaypee University of Information Technology
Waknaghat, Distt. Solan (Himachal Pradesh)