|  |  |  |
| --- | --- | --- |
| INTERNET OF THINGS BASED VOICE CONTROLLED WHEELCHAIR | | |
|  | | |
| PRN 05519004 | | |
| By | | |
|  | ALBRIGHT MACHUMU | 15171055004 |
|  | GILTON B TETI | 15171055040 |
|  | HADIJA SHABANI | 15171055044 |
|  | | |
| A PROJECT REPORT | | |
| Submitted to the | | |
| FACULTY OF ENGINEERING AND TECHNOLOGY | | |
| *In partial fulfillment of the requirements* | | |
| *for the award of the degree* | | |
| of | | |
| BACHELOR OF ENGINEERING | | |
| IN | | |
| COMPUTER SCIENCE AND ENGINEERING | | |
|  | | |
| Under the guidance of | | |
| Mr. Thangaraj Ethilu | | |
| Assistant Lecturer | | |
|  | | |
| ST. JOSEPH UNIVERSITY IN TANZANIA | | |
| DMI-ST. JOSEPH COLLEGE OF ENGINEERING AND TECHNOLOGY | | |
| DAR ES SALAAM, TANZANIA | | |
|  | | |
| JULY 2019 | | |



ST. JOSEPH UNIVERSITY IN TANZANIA

CERTIFICATE OF THE GUIDE

Certified that this project report “INTERNET OF THINGS BASED VOICE CONTROLLED WHEELCHAIR” is the bonafide work of ALBRIGHT MACHUMU (15171055004), GILTON B TETI (15171055040) and HADIJA SHABANI (15171055044) who carried out the project work under my supervision. This Thesis work is original one and not submitted earlier for the award of any degree / diploma elsewhere.

|  |  |
| --- | --- |
| (Dr. Prabakaran Narayanan) | (Mr. Thangaraj Ethilu) |
| HOD/ CSISE | PROJECT GUIDE |
|  | DEPARTMENT OF CSISE |

Submitted to the project report in viva-voce Examination held on …………………. at St. Joseph College of Engineering and Technology.

|  |  |
| --- | --- |
| INTERNAL EXAMINER | EXTERNAL EXAMINER |

ACKNOWLEDGEMENT

We convey our deepest sense of gratitude of all those who helped us in bringing out this project.

We would like to thank the management of St. Joseph University College of Engineering and Technology, Tanzania for providing this opportunity to do a project work as a part of our curriculum.

We also thank our College Chancellor Rev. Dr. J. E. Arulraj for providing us the necessary facilities for the competition of our project work.

We are extremely grateful to our President of University Council Dr. T. X. A. Ananth and Director of Education Director of Education Dr. Iganatius A. Herman & Ag. Vice Chancellor (Academic) Prof. Innocent Francis Ngalinda for their constant support and encouragement

We also express our cordial gratitude to our Principal Prof. Eliab O. Opiyo, for his valuable guidance and motivation.

We are extremely grateful to the Head of the Department Dr.Prabakaran Narayanan, for his valuable guidance, motivation, timely and insightful technical discussions. We are immensely grateful for his constant encouragement, smooth approach throughout our project period and make this work possible.

We are indebted to our Project Coordinator Mr. Thangaraj Ethilu, for his valuable suggestions and motivation. We are deeply indebted to our Internal Guide Mr. Thangaraj Ethilu faculty of Department of Computer Science and Engineering for extending her warm support, constant encouragement and ideas she shared with us.

We would be failing in our part if we do not acknowledge our family members and our friends for their constant encouragement and support.

ABSTRACT

Locomotion is one the basic needs of humans which is lacked by people having limited abilities due to several reasons such as accidents, paralysis and others hence it renders them to use wheelchair for moving from one place to another. The purpose of this project is to serve these people by automating the process of moving in any direction by using Voice recognition. The wheelchair circuit consists of a RF and Bluetooth receiver used to receive these commands, then these commands are processed and interpreted using the Arduino then LED’s will light be indicating the type of movement direction of the motors and then motion of wheelchair motors is executed in order to achieve desired movement. This allows the disabled person operate the wheelchair easily. So, in this way, the wheelchair system is taken to a new level by the usage of automation and safety.

TABLE OF CONTENT

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| CHAPTER NO | TITLE | | | | | | PAGE NO |
|  | CERTIFICATE OF THE GUIDE | | | | | | i |
|  | ACKNOWLEDGMENT | | | | | | ii |
|  | ABSTRACT | | | | | | iii |
|  | PROFORMA FOR APPROVAL OF PROJECT PROPOSAL | | | | | | vii |
|  | BIO-DATA OF THE PROPOSED GUIDE FOR PROJECT | | | | | | viii |
|  | DECLARATION BY THE CANDIDATES | | | | | | ix |
|  | LIST OF FIGURES | | | | | | x |
|  | LIST OF TABLES | | | | | | xii |
|  | LIST OF SYMBOLS AND ABBREVATION | | | | | | xiii |
| 1 | INTRODUCTION | | | | | | 1 |
|  | 1.1 | | PROBLEM STATEMENT | | | | 3 |
|  | 1.2 | | OBJECTIVE OF THE PROJECT | | | | 3 |
|  | 1.3 | | WHAT IS A WHEELCHAIR | | | | 4 |
|  | 1.4 | | WHY USE VOICE CONTROLLED WHEELCHAIR | | | | 5 |
| 2 | SYSTEM STUDY | | | | | | 6 |
|  | 2.1 | | REVIEW OF LITERATURE | | | | 6 |
|  | 2.2 | | FEASIBILITY STUDY | | | | 8 |
|  |  | | 2.2.1 | Technical Feasibility | | | 8 |
|  |  | | 2.2.2 | Economical Feasibility | | | 9 |
|  |  | | 2.2.3 | Operational Feasibility | | | 9 |
|  |  | | 2.2.4 | Schedule Feasibility | | | 9 |
| 3 | SYSTEM ANALYSIS AND REQUIREMENT | | | | | | 10 |
|  | 3.1 | | DESCRIPTION OF THE EXISTING SYSTEM | | | | 10 |
|  | 3.2 | | PROBLEM OF EXISTING SYSTEM | | | | 13 |
|  | 3.3 | | RESOURCE REQUIREMENTS | | | | 13 |
|  |  | | 3.3.1 | Hardware Requirements | | | 13 |
|  | 3.4 | | RESOURCE REQUIREMENT | | | | 13 |
|  |  | | 3.4.1 | Hardware Requirement | | | 13 |
|  |  | | 3.4.2 | Software Requirement | | | 19 |
| 4 | SYSTEM DESIGN | | | | | | 21 |
|  | 4.1 | | SYSTEM ARCHITECTURE | | | | 21 |
|  | 4.2 | | UML DIAGRAMS | | | | 22 |
|  |  | | 4.2.1 | Block Diagram | | | 22 |
|  |  | | 4.2.2 | Flow Chart | | | 23 |
|  |  | | 4.2.3 | Specifications | | | 24 |
| 5 | SYSTEM IMPLEMENTATION | | | | | | 25 |
|  | 5.1 | | MODULES DESCRIPTION | | | | 26 |
|  |  | | 5.1.1 | Voice input | | | 26 |
|  |  | | 5.1.2 | Voice processing circuit | | | 26 |
|  |  | | 5.1.3 | Arduino Circuit | | | 26 |
|  |  | | 5.1.4 | Motor driving circuit | | | 26 |
|  | 5.2 | | BLOCK DIAGRAM | | | | 27 |
|  | 5.3 | | DESCRIPTION OF THE BLOCK DIAGRAM | | | | 27 |
|  | 5.8 | | SIMULATION | | | | 29 |
| 6 | SYSTEM TESTING | | | | | | 30 |
|  | 6.1 | | SOFTWARE TESTING | | | | 30 |
|  | 6.2 | | LEVELS OF TESTING | | | | 30 |
|  |  | | 6.2.1 | | Unit Testing | | 31 |
|  |  | | 6.2.2 | | Integration Testing | | 32 |
|  |  | | 6.2.3 | | System Testing | | 32 |
|  |  | | 6.2.4 | | Acceptance Testing | | 33 |
|  |  | | 6.2.9 | | Non functional Testing | | 33 |
|  | 6.3 | | TET CASES FOR MODULES | | | | 34 |
|  |  | | 6.3.1 | | Test Case for LCD and Motors | | 34 |
|  |  | |  | | 6.3.1.1 | Test Case for LCD module | 34 |
|  |  | |  | | 6.3.1.2 | Test Case for Motors module | 35 |
|  |  | |  | | 6.3.1.3 | Test Case for Ultrasonic module | 37 |
|  |  | |  | | 6.3.1.4 | Test Case for Bluetooth module | 38 |
| 7 | CONCLUSION& FUTURE ENHANCEMENT | | | | | | 40 |
|  | 7.1 | | CONCLUSION | | | | 40 |
|  | 7.2 | | FUTURE ENHANCEMENT | | | | 40 |
|  | APPENDIX | | | | | | 41 |
|  | A1 | SAMPLE CODES | | | | | 41 |
|  | A2 | SCREENSHOTS | | | | | 46 |
|  | REFERENCES | | | | | | 48 |

PROFORMA FOR APPROVAL OF PROJECT PROPOSAL

Proposed project by:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Reg. No. |  | Name of the student | Semester | Branch |
|  |  |  |  |  |  |
|  | 15171055004 | Albright Machumu | | VIII | CSE |
|  |  |  |  |  |  |
|  | 15171055040 | Gilton B Teti | | VIII | CSE |
|  |  |  |  |  |  |
|  | 15171055044 | HadijaShabani | | VIII | CSE |
|  |  |  |  |  |  |
|  |  | |  | | |
|  | Title of the project: | | Internet Of Things Based Voice Controlled Wheelchair | | |
|  |  |  |  |  |  |
|  | Subject Area: |  | Internet of Things |  |  |
|  |  | |  |  |  |
|  | Name of the guide: | | Mr. Thangaraj Ethilu |  |  |
|  |  |  |  |  |  |
|  | Designation : |  | Assistant Lecturer |  |  |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| …………..……………… | | |  | ………………………………. | |
| (Mr. Albright Machumu) | | |  | (Mr. Thangaraj Ethilu) | |
| …………………………. | | |  |  |  |
| (Mr. Gilton B Teti) | | |  |  |  |
| …………………………. | | |  |  |  |
| (Miss. Hadija Shabani) | | |  |  |  |
| For Office Use only: | | |  |  |  |
|  |  |  |  |  |  |
|  | SYNOPSIS | APPROVED |  | NOT APPROVED |  |
|  |  |  |  |  |  |
|  | GUIDE | APPROVED |  | NOT APPROVED |  |
|  |  |  |  |  |  |

Date: (Dr. Prabakaran Narayanan)

HOD / CSISE

BIO-DATA OF THE PROPOSED GUIDE FOR PROJECT

1. PERSONAL INFORMATION

|  |  |  |
| --- | --- | --- |
| Name (in Block letters) | : | Mr. Thangaraj Ethilu |
| Sex | : | MALE |
| Academic Qualification | : | M.Tech(CSE) |
| Official Address | : | St. Joseph college of Engineering and Technology, P.O. Box:11007, Dar Es Salaam, Tanzania |
| Phone No / Fax | : | +255 683736795 |
| Residential Address | : | P.O. Box 11007, Dar Es Salaam |
| Phone/Mobile No | : | +255 683736795 |
| E-mail ID | : | [ethilthangaraj@yahoo.co.in](mailto:ethilthangaraj@yahoo.co.in) |

2. DETAILS OF EMPLOYMENT

|  |  |  |
| --- | --- | --- |
| Designation | : | Assistant Lecturer |
| Field of Specialization | : | Computer Science |
| Teaching Experience | : | 8 years |
| Industrial Experience (in years) | : | 5 years |
| Experience in the field of specialization | : | Networking |
| No. of Projects guided | : | 60 |

I, Mr.Thangaraj Ethilu, do hereby accept to guide Mr. Albright Machumu, Gilton B Teti and Miss Hadija Shabani, the student of the degree in Computer Science And Engineering programme of St. Joseph College of Engineering and Technology, Tanzania.

………………………

(Mr. Thangaraj Ethilu)

DECLARATION BY THE CANDIDATE

Certified that this project report “INTERNET OF THINGS BASED VOICE CONTROLLED WHEELCHAIR” is the bonafide work of“ALBRIGHT MACHUMU (15171055004), GILTON B TETI (15171055040) and HADIJA SHABANI (15171055044)” in partial fulfillment of seven semester degree of Computer Science and Engineering of St. Joseph college of Engineering and Technology, Tanzania under the guidance of Mr. Thangaraj Ethilu. This Thesis work is original one and not submitted earlier for the award of any degree / diploma elsewhere.

…………..………………

(ALBRIGHT MACHUMU)

………………………….

(GILTON B TETI)

………………………….

(HADIJA SHABANI)

LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| FIG NO | FIG NAME | PAGE NO |
| 1.1 | Wheelchair | 4 |
| 2.1 | Ordinary Wheelchair | 6 |
| 2.2 | Joystick Electric Wheelchair | 6 |
| 3.1 | Voice Recognition Chip | 13 |
| 3.2 | Microphone | 14 |
| 3.3 | Android Mobile Phone | 15 |
| 3.4 | Dc Motors | 16 |
| 3.5 | Arduino UNO | 17 |
| 3.6 | Bluetooth Module HC-05 | 18 |
| 3.7 | Arduino IDE | 19 |
| 4.1 | System Architecture | 20 |
| 4.2 | Block Diagram | 21 |
| 4.3 | Data flow | 22 |
| 5.1 | Block Diagram | 25 |
| 5.2 | Simulation Done Using Proteus Software | 29 |
| 6.1 | Levels of Testing | 31 |
| A2.1 | Home Screen | 46 |
| A2.2 | Home Activity | 46 |
| A2.3 | Navigation Bar | 47 |
| A2.4 | Touch Activity | 47 |
| A2.5 | Motor Driver | 48 |
| A2.6 | Ultrasonic Sensor | 48 |
| A2.7 | Atmega38p Chip | 48 |
| A.2.8 | LCD | 48 |

LIST OF TABLES

|  |  |  |
| --- | --- | --- |
| **TAB. NO** | **TABLE NAME** | **PAGE NO** |
| 6.1 | LCD table | 34 |
| 6.2 | Motors table | 35 |
| 6.3 | Ultrasonic Sensor Table | 37 |
| 6.4 | Bluetooth Transmission Table | 38 |

LIST OF ABBREVIATIONS

|  |  |
| --- | --- |
| ACRONYM | ABBREVIATION |
| MS | Multiple Sclerosis |
| PC | Personal Computer |
| IC | Integrated Circuit |
| CPU | Central Processing Unit |
| RAM | Random Access Memory |
| APK | Application Programming Kit |
| SDK | Software Developer’s Kit |
| IDE | Integrated Development Environment |
| PCB | Printed Circuit Board |
| IOT | Internet Of Things |
| IJRET | International Journal Of Research in Engineering and Technology |
| ADL | Activities Of Daily Living |
| DC | Direct Current |
| USB | Universal Serial Bus |
| MHz | Mega Hertz |
| ICSP | In Circuit Serial Programming |
| EDR | Enhanced Data Rate |
| AFH | Adaptive Frequency Hopping |
| SPP | Serial Port Protocol |
| GPL | General Public License |
| LCD | Liquid Crystal Display |
| GPS | Global Positioning System |

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Currently around 10% of the total world‘s population, or roughly 650million people live with a disability. Every year, ten thousand cases added to this group. The number of people, who need to move around with the help of some artificial means, whether through an illness or an accident, is continually increasing. These means, the number of disabled people is increasing by tragic accidents also, some victims of the accidents are suffering from abnormal life with serious spinal injuries. Freedom of mobility is a dream for every person with physical disabilities.

With the aid of technological evolution, in order to increase the quality of life for these people and facilitate their integration into the working world. In this way a contribution may be made to facilitating Wheelchair is widely used device by disabilities. While the needs of many individuals with disabilities can be satisfied with traditional manual or powered wheelchairs, a segment of the disabled community finds it difficult or impossible to use wheelchairs independently physically challenged people have either less mobility orno mobility. The various paralyses such as quadriplegia and Multiple sclerosis. Multiple sclerosis. Multiple sclerosis (MS), also known as disseminated sclerosis or encephalomyelitis disseminate, is an inflammatory disease in which the insulating covers of nerve cells in the brain and spinal cord are damaged. This damage disrupts the ability of parts of the nervous system to communicate, resulting in a wide range of signs and symptoms, including physical, mental and sometimes psychiatric problems.

The name multiple sclerosis refers to scars (sclera—better known as plaques or lesions) particularly in the white matter of the brain and spinal cord.MS takes several forms, with new symptoms either occurring in isolated attacks (relapsing forms) or building up over time (progressive forms). Between attacks, symptoms may go away completely; however, permanent neurological problems often occur, especially as the disease advances. There is no known cure for multiple sclerosis. In severe cases the patient becomes paralyzed, while in milder cases there may be numbness in the limbs. Quadriplegia is paralysis caused by illness or injury to a human that results in the partial or total loss of use of their entire limb.

Quadriplegia is caused by damage to the brain or the spinal cord at a high level C1–C7—in particular, spinal cord injuries secondary to an injury to the cervical spine. The injury, which is known as a lesion, causes victims to lose partial or total function of all four limbs, meaning the arms and the legs. Quadriplegia is defined in many ways; C1–C4 usually affects arm movement more so than a C5–C7 injury; however, all quadriplegics have or have had some kind of finger dysfunction. So, it is not uncommon to have a quadriplegic with fully functional arms but no nervous control of their fingers and thumbs.

Typical causes of this damage are trauma (such as a traffic collision, diving into shallow water, a fall, a sports injury), disease (such as transverse mellitus, multiple sclerosis, or polio), or congenital disorders (such as muscular dystrophy).Some quadriplegic individuals can walk and use their hands, as though they did not have a spinal cord injury, while others may use wheelchairs and they can still have function of their arms and mild finger movement, again, that varies on the spinal cord damage. These results indicate a need for entirely new technology for the independent mobility of such patients. The normal electric powered wheelchair needs to be operated through joystick by hand. Since these patients are incapable for using a manual or electrically powered wheelchair the ideal solution is to use source such as voice. The project aims at designing and developing a microcontroller-based voice controlled electric wheelchair with a voice recognition module to drive the wheelchair using voice commands. Here we are going to control the wheelchair by using the voice commands of the user. The user can easily learn and adapt the working of this wheelchair with minimum effort and time.

1.2 PROBLEM STATEMENT

According to World report on disability" jointly presented by World Health Organization (WHO) and World Bank says that there are 70 million people are handicapped in the world. Unfortunately, day by day the number of handicapped people is going on increasing due to road accidents as well as disease like paralysis. If a person is handicapped, he is dependent on other person for his day to day work like transport, food, orientation etc. So, a voice operated wheel chair is developed which will operate automatically on the commands from the handicapped user for movement purpose. The following are the scenarios in which this wheelchair will be applicable:-

1. Due to accidents and injury

Most of disabled are in this category such that due to several accidents like car,motocyle and other accidents may lead to the disability of locomotion

1. Brain damages such as multiple sclerosis

This is a neurological disease that may lead to paralysis and hence a patient may loose his/her locomoting ability

1.3 OBJECTIVES OF PROJECT.

In developing this system, some project objectives had been specified. The main purpose of this project is to improve the current existing wheelchair used by the disabled by developing an IOT based wheelchair Some objectives of this project had been identified and listed below.

1. To replace the current existing wheelchair to fully computerized and automated system.
2. To develop a mobile-based application that obtains the voice input from the user to direct the wheelchair to the specified direction.

1.4 WHAT IS A WHEELCHAIR?

A wheelchair is a chair with wheels, used when walking is difficult or impossible due to illness, injury, or disability. Wheelchairs come in a wide variety of formats to meet the specific needs of their users. They may include specialized seating adaptions, individualized controls, and may be specific to particular activities, as seen with sports wheelchairs and beach wheelchairs. The most widely recognized distinction is between powered wheelchairs ("powerchairs"), where propulsion is provided by batteries and electric motors, and manually propelled wheelchairs, where the propulsive force is provided either by the wheelchair user/occupant pushing the wheelchair by hand ("self-propelled"), or by an attendant pushing from the rear ("attendant propelled").



Fig. 1.1Wheelchair

1.5 WHY USE ELECTRONIC VOICE CONTROLLED WHEELCHAIR?

Minimal effort to mobilize: Minimal effort is needed to control the wheelchair because you use speech.

Independence: With an electric wheelchair, you don't have to depend on someone else to push the chair, allowing you the independence of moving about.

Adaptable for severity of disability: The adaptability of an electric wheelchair is astounding. People who have only control of their mouth and breathing have been set up to move their wheelchair using special controls, for example.

Going up hills and ramps: The power of an electric wheelchair makes it easier for you to go up or down hills.

Power for distance: When using a manual chair to go a far, you need to take into account how much energy and ability you have to come back. With an electric wheelchair, you are only limited by the amount of power left in your battery pack.

CHAPTER 2

SYSTEM STUDY

2.1 REVIEW OF LITERATURE

A literature review can be referring to as a review of current system that the researcher had done previously and the review of the system that will be developed. Literature review also focuses on the knowledge and ideas established on a topic as well as their strengths and weaknesses. Nowadays, technology is getting better and better to replacing the traditional system to speed up the process by introducing the computerized system. There are few types of wheelchairs that had been introduced nowadays.

There are many scientists and researchers who develop computer software that can recognize human voice commands in so many languages such as English, Japanese and Thai.

There are many techniques that are used to recognize voice commands Researchers transform sound wave into digital wave by a computer. After that they use digital signal to manage different electronic equipments, for example

1)controlling robot arm movement

2)helping the handicapped to move a wheel chair etc.

According to “ IJRET” In the paper on “Voice Operated Intelligent Wheelchair” , Mat lab software is used for input signal processing and that signal given to the ARM Processor LPC2138.

In recent paper of “ IJRET”, input is given to IC HM2007.HM 2007 IC is used for the voice recognition purpose. HM 2007 generates the output signal depending on the input from the user.



Fig. 2.1 Ordinary wheelchair



Fig. 2.2 Joystick Electric wheelchair

2.2 FEASIBILITY STUDY

An analysis and evaluation of a proposed project to determine if it is technically feasible or technically feasibility, is feasible within the estimated cost or economic feasibility, and will be profitable. Feasibility studies are almost always conducted where large sums are at stake. Also called feasibility analysis. See also cost benefit analysis.

Feasibility analysis begins once the goals are defined. It starts by generating broad possible solutions, which are possible to give an indication of what the new system should look like. This is where creativity and imaginations are used. Analysts must think up a new ways of doing things and generate new ideas. There is no need to go into detailed system operation yet. The solution should provide enough information to make reasonable estimate about project cost and give users an indication of how the new system will fit into the organization. It is important not to exert considerable effort at this stage only to find out that the project is not a worthwhile or that there is a need significantly change the original goal.

Feasibility of the new system means ensuring that the new system, which we are going to implement, is efficient and affordable. There are various types of feasibility to be determined. They are

2.2.1 Technical Feasibility

The technical requirements for the system are economical and it doesn’t use any additional hardware and software for its functionality. Technical evaluation must also assess whether the existing system can be upgraded to use the new technology and whether the organization has the expertise to use it.

The system working is quite easy to use and has a gentle slope due to its simple but attractive user interface. User requires no special training for operating system.

2.2.2 Economical Feasibility

The system being developed is economically feasible with respect to needy point of view. It is cost effective in the sense that has eliminated the energy input into pushing the chair thus the only thing to be done is making an environment with an effective supervision.

2.2.3 Operation Feasibility

The system working is quite easy to use and learn due to its simple but attractive interface. User requires no special training for operating the system. Technical performance includes issues such as determining whether the system can provide the right response so that it always delivers this output at the right direction and on time.

2.2.4 Schedule Feasibility

Schedule Feasibility is defined as the probability of a project to be completed within its scheduled time limits, by a planned due date. If a project has a high probability to be completed on-time, then its schedule feasibility is appraised as high. In many cases a project will be unsuccessful if it takes longer than it was estimated: some external environmental conditions may change; hence a project can lose its benefits, expediency and profitability. If a work to be accomplished at a project does not fit the timeframes demanded by its customers, then a schedule is unfeasible (amount of work should be reduced or other schedule compression methods applied).

If the project managers want to see their projects completed before they can lose their utility, they (project managers) need to give proper attention to controlling their schedule feasibility: to calculate and continually reexamine whether it is possible to complete all amount and scope of

CHAPTER 3

SYSTEM ANALYSIS AND REQUIREMENTS

3.1 DESCRIPTION OF THE EXISTING SYSTEM

Wheelchairs have evolved very little over the past 1000 years. Most of the design changes have occurred within recent decades. In 6th Century A.D Earliest recording of a wheelchair was found that a Chinese engraving picturing a man in a chair with three wheels. Later in 16th Century A.D, Wheelchairs were well-developed in Europe and commonly found in drawings and literature.

In 1869 the first wheelchair patent was issued in the United States. Late 1903an electrically-driven wheelchair operating on a 12-volt battery and a 3/8 horsepower motor was used to give people rides. At the time it was not used for handicapped mobility but it did pave the way for future developments. During World War I, the first electric wheelchairs were used for the handicapped. A battery and motor were applied to existing wheelchairs with a simple one-speed on/off switch. At last 1940 the first patent was issued for an electric wheelchair. In 1950 Sam duke received a patent for a releasable add-on power drive applied to manual wheelchair (the unit was actually permanently fitted to the chair with Unbolts).

At late 1990‘s, the popular electric wheelchairs on the market are foldable though they require removal of at least the leg rests and batteries. Folding, Light-Weight wheelchairs, manual and powered models: These wheelchairs are usually compact for indoor use and have small, self-pivoting wheels (casters) in front. Most of these powered wheelchair models are controlled by the user’s hand with a small control stick. Quadriplegics unable to use a hand for steering and control. These foldable power wheelchair models are usually purchased because they can be folded for storage, for transport in a car or car-trunk, or for transport on a train or airplane.

Light-Weight ―Power-Assisted wheelchair models:

These are usually similar to folding manual wheelchairs and are manually propelled in similar ways. These models do have a battery plus one or two motors for propulsion assistance. Typically, the user is a paraplegic or quadriplegic person who desires to manually propel himself as much as possible for the exercise and resulting health benefits. These wheelchair models are often lighter-weight than the folding power models discussed above because the intermittent power use allows for smaller battery and motor sizes.

Non-Folding, Stand-Up power wheelchair models:

These are usually sometimes used by people, unable to stand unaided, so that they can be stood up by the wheelchair for household chores or to converse face-to-face with non-handicapped people who are also standing. When in the elevated position, this type wheelchair has a very high center-of-gravity and can easily topple over. Stand-up wheelchairs are safe only on perfectly flat and smooth flooring and only if the user is properly strapped to the seat and seat back.

Combination ―Indoor-Outdoor‖ power wheelchair:

These models are often purchased by people able to have only a single power wheelchair. Lack of storage space or limited finances can prevent the safer option: a person owns two power models – one for optimal indoor use – and another for optimal outdoor use. Outdoor safety requirements usually require that large, powered wheels be in front. These ―combo‖ wheelchairs are probably most useful and safe when used (1) outdoors, on pavements and sidewalks in urban and suburban areas – not on grass or soil in rural areas; and (2) indoors, in nursing homes, assisted living facilities or apartments with wider halls and doorways that facilitate the wheelchair with larger turning radius and maneuvering space requirements.

Outdoor power wheelchair models.

These are Very efficient outdoors, most have large diameter wheels in front with which to climb and steering may be accomplished, depending on design, by (1) rotating a rear-wheel dolly or (2) powering one of the large front wheels while breaking the other. Many such power wheelchairs are able to safely climb grades of as much as 40%; climb curbs and, sometimes, climb outdoor stairs as seen in front of a courthouse or public building. Often, these power wheelchairs are not safe or practical for use in a particular apartment. The type of artificial aid needed by a disabled person in order to move depends to a large extend on the level of incapacity. If the user is capable of controlling head or hands, the ideal solution is the use of a joystick where there is a high level of incapacity; solution is basically centered on use of another mean, such as the voice. In the case of multiple sclerosis (MS) and quadriplegics the mobility is completely absent or restricted. People with spinal cord injuries, quadriplegic patients, multiple sclerosis are living a very painful life. Their families and their friends are also suffering from taking continuous care of the patients. A basic instinctive of life, moving around is an evidence of what they are still alive. These patients also have the right to move and want to move at all costs.

Several studies have concluded that the independent mobility or movement which is included powered wheel chair, manual wheelchair and walker access the benefit to all the disabled human beings. Independent mobility increases vocational and educational opportunities, reduces dependence on other members, and promotes feelings of self-reliance and in dependability.

Independent mobility plays a vital role in building the foundation for much early learning for young people. The lack of exploration and control often results into a cycle of deprivation and lack of motivation that leads to learned helplessness. For aged people, independent movement is an important aspect of self-esteem and plays a vital role in “aging in place.” Mobility difficulties led to the problem of activities of daily living (ADL) and instrumental ADL disabilities because of the need to move to accomplish many of these activities.

The impaired mobility often results in reduced opportunities to have socialized policies, which leads to social isolation, and many mental problems. While the needs of many individuals with disabilities can be satisfied with traditional manual or self-automated wheelchairs, a segment of the disabled community finds it difficult or impossible to use wheelchairs independently.

The disabled population includes people with low vision, visual field reduction, spasticity, tremors, or cognitive deficits. These individuals depend on other people for mobility to push them in a manually handled wheelchair. To accommodate this population, several researchers have used technologies originally developed for Power wheelchairs have been designed of different ways, such as assuring collision free travel, aiding the performance of specific tasks (e.g., passing through doorways), and autonomously transporting the user between locations. The Idea of using voice-based technology for controlling the motion of the wheels of wheelchair is to prove that this project stands one step ahead of other average projects. The use of this new technology in conjunction with a mechanical system in order to simplify everyday life would spark interest in the developing modern society. Many people with disabilities do not have the dexterity necessary to control a joystick on an electrical wheelchair.

3.2 PROBLEM OF THE EXISTING SYSTEM

Here are some of the typical daily challenges faced by wheelchair users:

1. Wheelchair users don’t have a choice but to sit which can present the problems of muscle cramps and pressure sores. Either affliction can be a great source of discomfort.
2. Dependence: With an ordinary wheelchair, the user is depending to the external person to support and push the wheelchair.
3. Adaptable for severity of disability: The adaptability of an electric wheelchair is astounding. People who have only control of their mouth and breathing have been set up to move their wheelchair using special controls, for example. Joystick,
4. Going up hills and ramps: The power of an electric wheelchair makes it easier for you to go up or down hills.

3.3 JUSTIFICATION OF THE PROPOSED SYSTEM

In this work, voice recognition module is used as user interface. This wheelchair is driven and fully controlled by using voice commands. In which the voice commands of the user are recognized by the voice identification module. The digital output obtained from the voice recognition module is used to drive Arduino based control circuit. Arduino is programmed in such a way to produce the required output for the corresponding voice commands. The output of the Arduino is given to a motor driver, which can control a set of 2 dc motors simultaneously in any direction.

3.3 RESOURCE REQUIREMENTS

3.3.1 Hardware Requirements

-Arduino UNO

-L293D Motor Shield

-Bluetooth Module HC-05

-Ultrasonic Sensor HC-SR-04

-Wheel chair chassis

-Battery 12 volts

-2 DC Motors (12 V 200 rpm)

-Android mobile phone

-Power supply

a. Voice Recognition Chip

It is the heart of the entire system. HM2007 is a voice recognition chip with on-chip analog front end, voice analysis, recognition process and system control functions. The input voice command is analyzed, processed, recognized and then obtained at one of its output ports which is then decoded, amplified and given to motors of robot. If the computer's CPU time. When the HM2007 recognizes a command, it can signal an interrupt to the host CPU and then relay the command code. The HM2007 chip can be cascaded to provide a larger word recognition library. The circuit we are building operates in the manual mode. The manual mode allows one to build a standalone speech recognition board that doesn't require a host computer and may be integrated into other devices to utilize speech control.

The HM2007 is a single-chip complementary metal-oxide semiconductor (CMOS) voice recognition large-scale integration (LSI) circuit. The chip contains an analog front-end voice analysis, recognition, and system control functions. The chip may be used in a stand-alone or connected.



Fig. 3.1 Voice Recognition chip

Features:

1. Single-chip voice-recognition CMOS LSI.
2. Speaker-dependent
3. External RAM support
4. Maximum of 40-word recognition
5. Maximum word length of 1.92 s
6. Microphone support
7. Manual and CPU modes available
8. Response time less than 300 milliseconds (ms)
9. 5-volt (5V) power supply

b. Microphone

It takes the analog voice commands and sends it to voice recognition chip (HM 2007) in the form of electrical signal. The human ear has an auditory range from 10 to 15,000 Hz. Sound can be picked up easily using a microphone and amplifier. This voice-controlled wheelchair uses unilateral mic. The unilateral mic is capable of ignoring noises apart from the actual voice commands. The mic receives the voice commands from the user and send it to the voice recognition module



Fig. 3.2. Microphone

c. Android Mobile phone

This is used to display the voice input as text and feed the output to the voice recognition circuit.



Fig. 3.3 Android Mobile Phone

d. D.C Motor

The Arduino is coupled with motors to drive the system as per the user’s demand. The proposed system consists of four, DC magnetized, 100 rpm motors (Johnson motors) to equalize the weight distribution and stall torque. The motors will be attached to all the four wheels and two motor drivers will be connected to each side.



Fig. 3.4. DC Motors

e. Arduino Uno

The Arduino Uno R3 acts as an intermediate agent between the voice recognition module and the motors to drive the wheelchair. It is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins, 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller. It receives the input given to the voice recognition module and converts into the format accepted by the motors and thus the motors work according to the command given. The Arduino needs to be interfaced to the motors as well as the voice control module.



Fig. 3.5. Arduino UNO

f. Bluetooth Module HC-05

HC‐05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. The HC-05 Bluetooth Module can be used in a Master or Slave configuration, making it a great solution for wireless communication. This serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate)3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Blue core 04‐External single chip Bluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping Feature).



Fig.3.6Bluetooth Module HC-05

3.3.2 Software Requirements

-Arduino IDE

-Android Application

-Coding language: Arduino programming (java & C++)

Arduino IDE

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It is used to write and upload programs to Arduino board. Arduino IDE is an open-source software program that allows users to write and upload code within a real-time work environment. The system is fully compatible with any Arduino software board.

The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub *main()* into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program *avrdude* to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

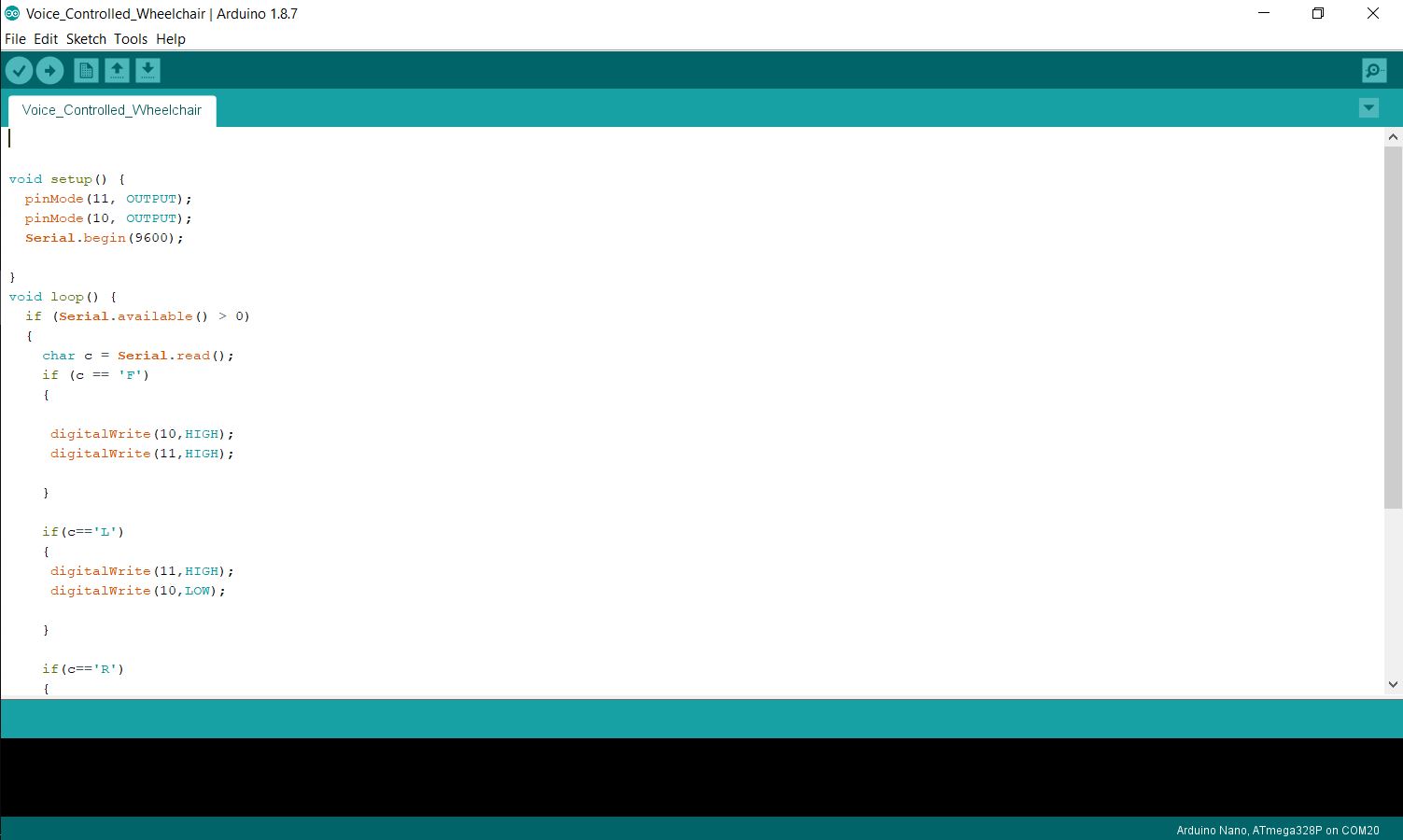


Fig. 3.7. Arduino IDE

CHAPTER 4

SYSTEM DESIGN

* 1. SYSTEM ARCHITECTURE

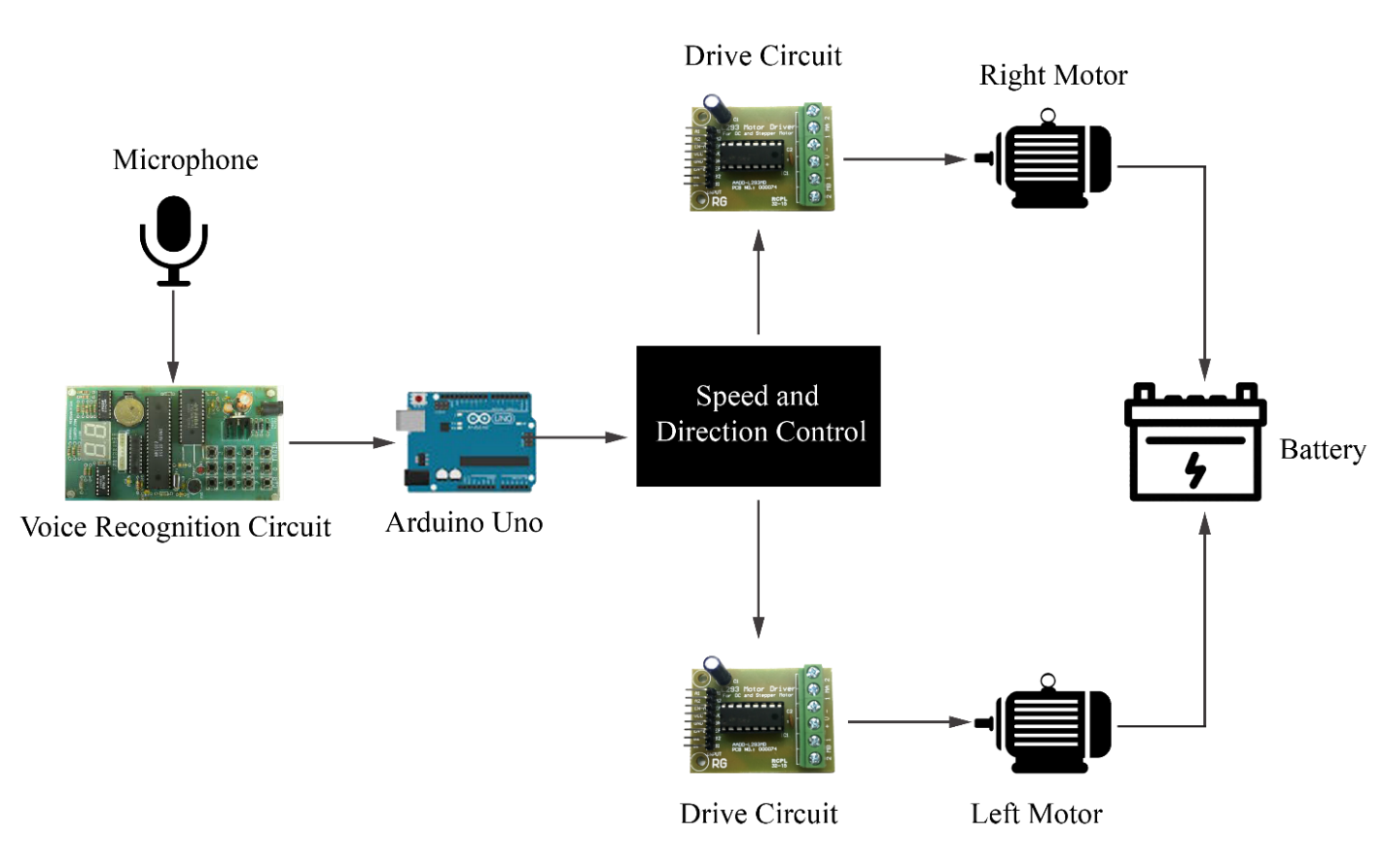


Fig.4.1 System Architecture

4.2 UML DIAGRAMS

4.2.1 Block Diagram

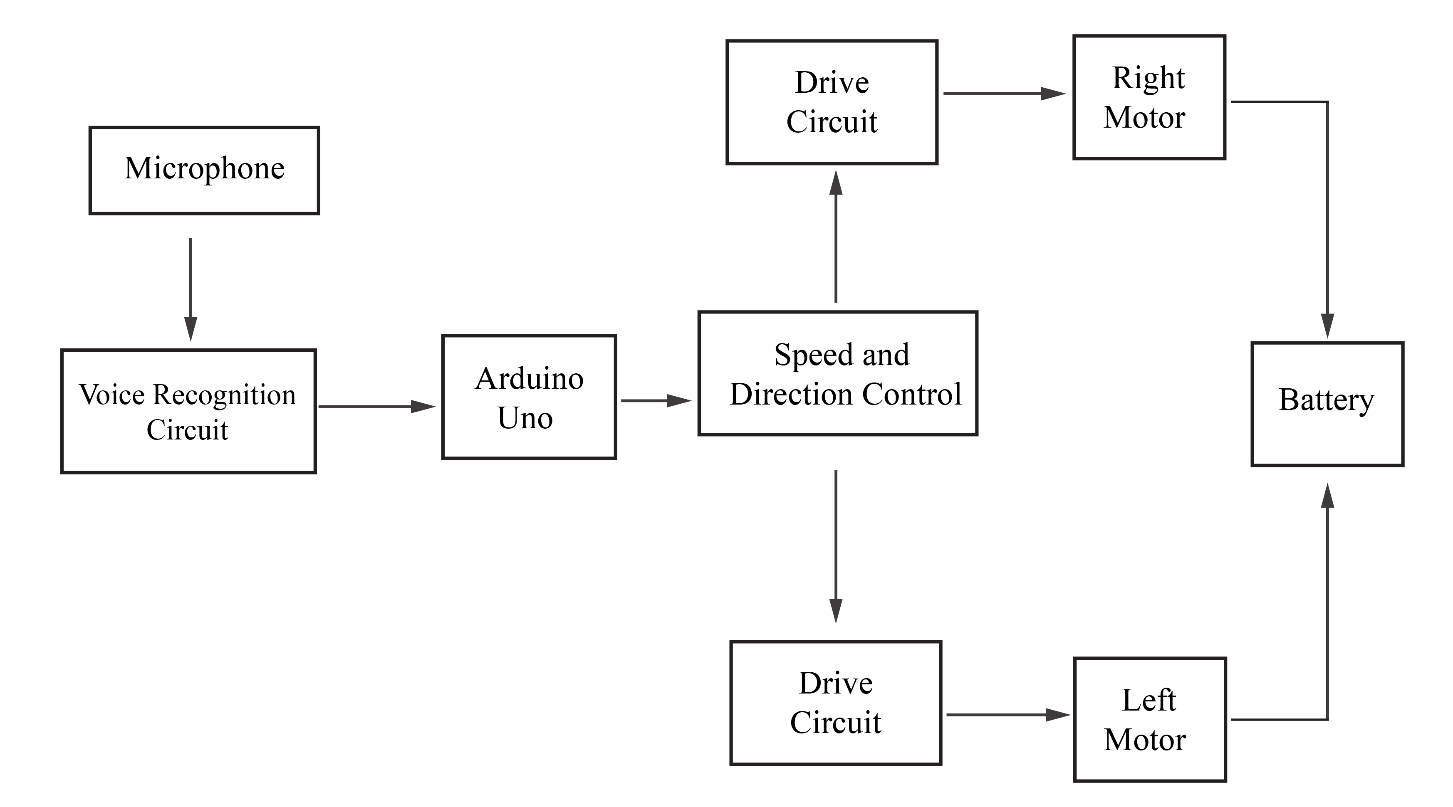


Fig. 4.2. Block Diagram

Above diagram shows the system block diagram showing the interconnections between each block or module of the system. All the modules are mounted onboard as to ease the wheelchair movement. This includes a microphone which is located nearest to the user so as to make it handy and easy to use. Generally, the input voice level of the user affects the recognition accuracy of the command given result. Principally, the system is triggered by the voice command word produced by the user through the use of this microphone. The voice of the user using the system is already trained and stored in the module. So, when the user gives the command, the module matches it with the existing command and gives the output if the voice and the command match.

The working of the wheel chair is based on the voice recognition unit which is the heart of the system. There are five types of motions considered, moving forward, moving in reverse direction, moving to the left and moving to the right and stop. The system starts by applying the supply voltage to the speech recognition circuit. The direction of the wheelchair depends on the user. For the forward command the wheelchair moves in forward direction. For the reverse direction the opposite movement of wheel rotation will occur. The left command is dependent on the mechanism of the wheel i.e. right wheel moves forward and left wheel moves backward and right command makes left wheel moves forward and right wheel rotate backward. However, all these commands are to be fed into the voice recognition kit via Android mobile phone. The wheelchair system will go back to the stand by condition or end the whole system by turning off the power supply of the speech recognition board.

4.2.2 Data Flow Diagram

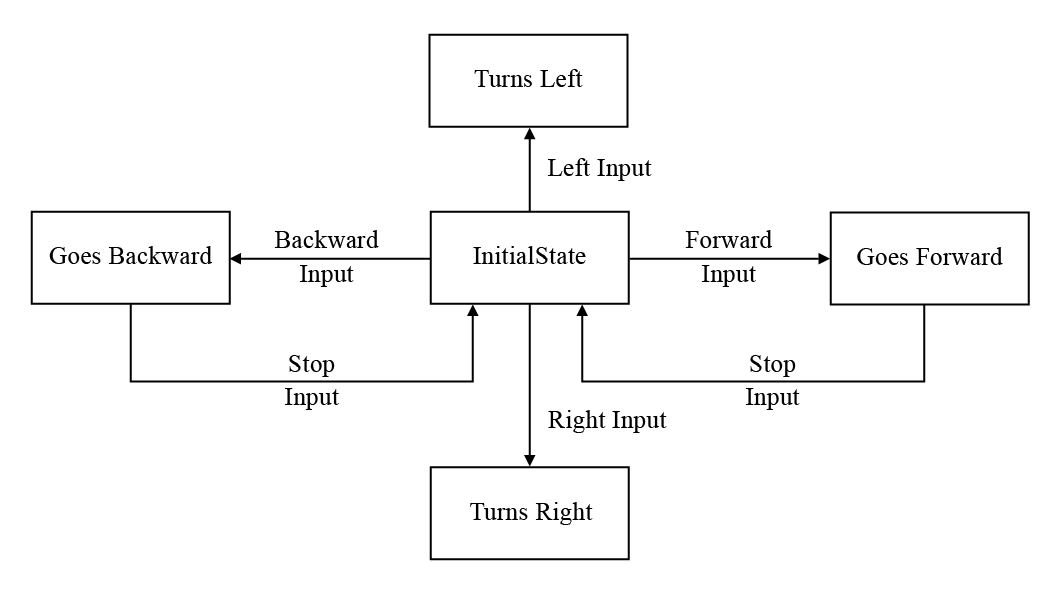


Fig. 4.3 Data flow

4.2.3 SPECIFICATIONS

HARDWARE: -

1.ATMEGA328P CHIP

2.BLUETOOTH MODULE HC-05

3. ULTRASONIC SENSOR SR

4.DC GEARED MOTORS

5.BATTERY 12V

CHAPTER 5

SYSTEM IMPLEMENTATION

5.0 INTRODUCTION

System implementation is the important stage of project when the theoretical design is tuned into practical system. The main stages in the implementation are as follows: -

1. Planning
2. Training
3. System testing and implementation

Changeover Planning is the first task in the system implementation. At the time of implementation of any system people from different departments and system analysis involve. They are confirmed to practical problem of controlling various activities of people outside their own data processing departments. The line managers controlled through an implementation coordinating committee. The committee considers ideas, problems and complaints of user department, it must also consider:

* The implication of system environment
* Self-selection and allocation for implementation tasks
* Consultation with unions and resources available

Standby facilities and channels of communication Student Attendance management system will implement student details, staff handle subject’s details, separate login details, and time table details. It will used to entered subject wise attendance. This application elaborates attendance table generate weekly, consolidate report provide to the End user. Mostly this application will calculate date wise attendance. To select starting date to end date generate reports at the time of activities.

5.1 MODULE DESCRIPTION

The secure project is made up of several modules which are integrated together in order to make the system’s functionality possible. These are modules which have been used in project completion:

1. Voice input
2. Voice processing
3. Arduino circuit
4. Motor driving circuit

5.1.1 Voice input

A user gives a voice input to the system which command a motion in a particular direction.

This is done by the use of the mic which acts as a device that takes the sound waves and convert them into electric signals.

5.1.2 Voice processing circuit

A voice input command is converted to a text input by the voice processing circuit.

5.1.3 Arduino Circuit

A text input from the voice processing circuit is fed to Arduino. This process the text to give the output to be fed to the motor driving circuit.

5.1.4 Motor driving circuit

This circuit drives the right and left motor as per command. It receives its input from the microcontroller.

5.2 BLOCK DIAGRAM

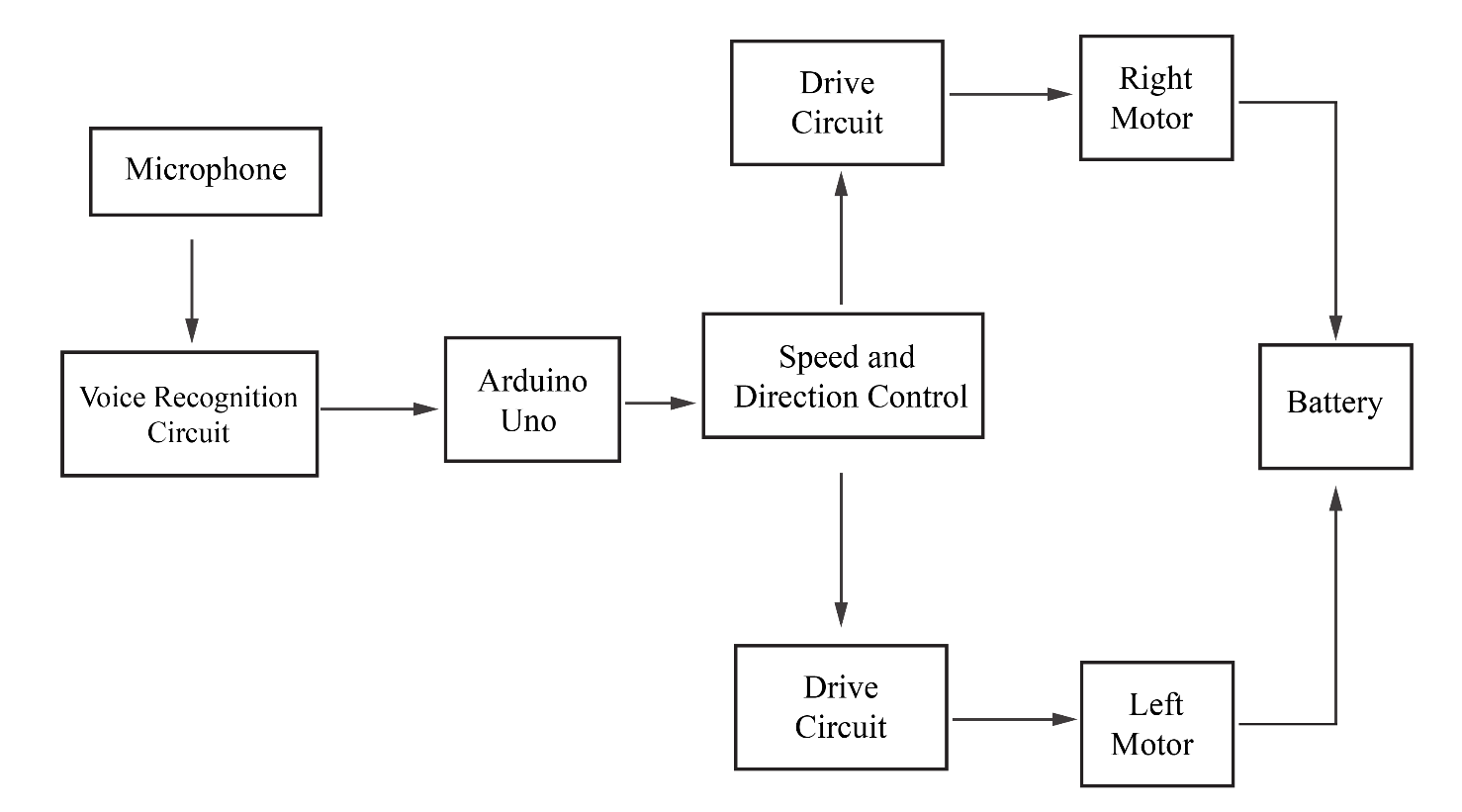


Fig. 5.1 Block diagram of a voice operated wheelchair

5.3 Description of the Block Diagram

HARDWARE:

Block diagram of voice operated wheelchair consist of following blocks.

1) PIC microcontroller

2)Voice recognition block

3) Driver IC block

4) DC motors block

5) Battery

6) Battery charger

The description of this blocks is as follows.

1)Microcontroller Pic16f877

This is a 40pin programmable interrupt microcontroller. It is a high-performance RISC CPU. This is used for controlling the movement and direction of wheelchair by controlling the two DC motors. The details of microcontroller are given in following section. The microcontroller unit is the core of intelligent wheelchair. It interfaces the voice recognition unit and the motor driver circuit. The main function of this unit is to receive the data from the HM2007 IC through (D0-D7) and determine the right command to be given to the driver circuit.PIC16F877Amicrocontroller with 33 I/O lines covers all the requisites for this wheelchair.

2)Voice Recognition IC Hm2007

The voice recognition unit consists of the HM2007 IC. It is a Large-Scale Integration (LSI) circuit with analog front end, voice analyzer, voice recognition processor and functional control system embedded in a single chip Complementary Metal Oxide Semiconductor (CMOS). It also consists of HM6264B IC which is a64k external static RAM used by the HM2007 IC to store the trained words that are used at the recognition phase, a 4\*3 keyboard, external microphone and some other components assembled together to build a 40 isolated sound word recognition system. The voice recognition IC HM2007 is operated in speaker dependent recognition mode. In this mode, the unit responds only to the current user. If another person needs to use the same system, a new training phase must be applied. This mode reaches a high accuracy of more than 95% for voice command recognition.

3) Motor Driver Circuit

The L293 and L293D are quadruple high-current half-H drivers. The L293 is designed to provide bidirectional drive currents of up to 1 A at voltages from 4.5 V to 36 V. The L293D is designed to provide bidirectional drive currents of up to600-mA at voltages from 4.5 V to 36 V. Both devices are designed to drive inductive loads such as relays, solenoids, dc and bipolar stepping motors, as well. as other high-current/high-voltage loads in positive-supply applications.

4)Motors(dc):

Two 12V dc motors are used in this experiment.

5)Power Supply Section

This section is consisting of a rechargeable battery. This section deals with the power requirements of the wheel chair for DC motors, Microcontroller and other section. Battery is used to provide the power supply to L298 driver IC (supply)which drives the DC motors, Microcontroller and IR section operates on 5V supply which is provided with the help of LM7805 which is a 5V regulator IC by converting 12V into 5V.

5.3 MOTION OF THE WHEEL CHAIR

The main part of the design is to control the motion of the wheelchair. There are four condition of motions are considered, moving forward, moving in reverse direction, moving to the left and moving to the right. For the speed, the user may use slow or fast speed command.

The starts by applying the supply voltage to the speech recognition circuit. For fast condition the system will supply higher current to the motors

If the user doesn’t want the wheelchair to move in high speed, the slow speed command can be set by applying low current supply to the motors, the wheelchair directions and movement possible are as given below

* Forward: Both motors are in forward direction
* Reverse: Both motors are in reverse direction
* Left: Right motor stopped and left motor in forward direction
* Stop: Both motors are stopped

5.4 SIMULATION

A **simulation** is an approximate imitation of the operation of a process or system;[]](https://en.wikipedia.org/wiki/Simulation#cite_note-definition-1) the act of simulating first requires a model is developed. This model is a well-defined description of the simulated subject, and represents its key characteristics, such as its behavior, functions and abstract or physical properties. The model represents the system itself, whereas the simulation represents its operation over time.

Simulation is used in many contexts, such as simulation of technology for performance optimization, safety engineering, testing, training, education, and video games. Often, computer experiments are used to study simulation models. Simulation is also used with scientific modelling of natural systems or human systems to gain insight into their functioning, as in economics. Simulation can be used to show the eventual real effects of alternative conditions and courses of action. Simulation is also used when the real system cannot be engaged, because it may not be accessible, or it may be dangerous or unacceptable to engage, or it is being designed but not yet built, or it may simply not exist.

Key issues in simulation include the acquisition of valid source information about the relevant selection of key characteristics and behaviors, the use of simplifying approximations and assumptions within the simulation, and fidelity and validity of the simulation outcomes. Procedures and protocols for model verification and validation are an ongoing field of academic study, refinement, research and development in simulations technology or practice, particularly in the field of computer simulation.

The following are some of the software used in simulation

* MATLAB
* PROTEUS
* SIMIO
* WITNESS
* FLOW 3D

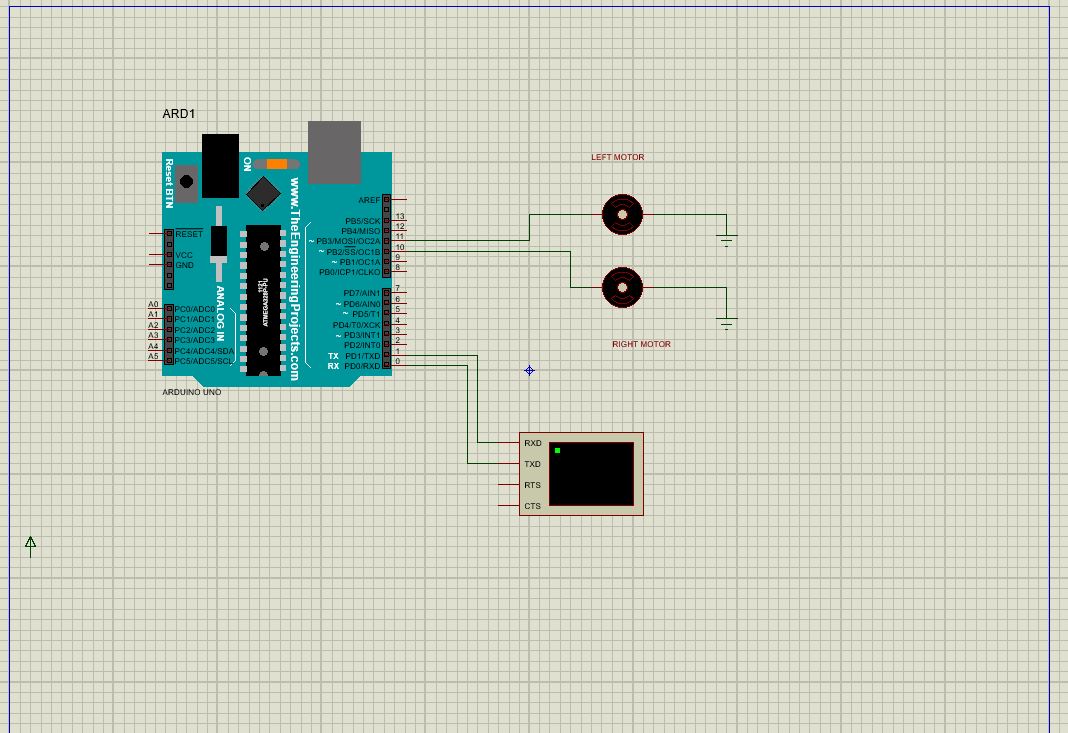


Fig 5.2 Simulation done using Proteus Software

CHAPTER 6

SYSTEM TESTING

6.1 SOFTWARE TESTING

As the project is on bit large scale, we always need testing to make it successful. If each component works properly in all respect and gives desired output for all kind of inputs then project is said to be successful. So, the conclusion is-to make the project successful, it needs to be tested.

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black-box testing, and as such, should require no knowledge of the inner design of the code or logic.

The testing done here was System Testing checking whether the user requirements were satisfied. The code for the new system has been written completely using Django which is the python framework for the backend and Ionic3 which is the angular 4 frameworks. The new system has been tested well with the help of the users and all the applications have been verified from every nook and corner of the user.

Although some applications were found to be erroneous these applications have been corrected before being implemented. The flow of the forms has been found to be very much in accordance with the actual flow of data.

6.2 LEVEL OF TESTING

There are generally four recognized levels of tests: unit/component testing, integration testing, system testing, and acceptance testing. Tests are frequently grouped by where they are added in the software development process, or by the level of specificity of the test.

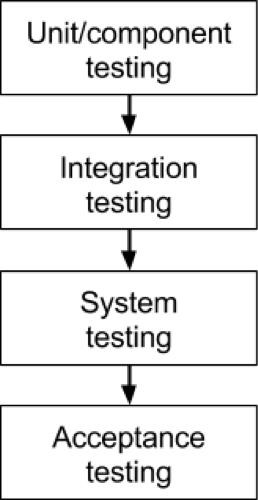


Fig.6.1 Levels of Testing

6.2.1 Unit Testing

This is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing.

Unit testing focuses verification efforts on the smallest unit of the software design, the module. This is also known as “Module Testing”. The modules are tested separately. This testing carried out during programming stage itself. In this testing each module is found to be working satisfactorily as regards to the expected output from the module.

6.2.2 Integration Testing

This is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.

Data can be grossed across an interface; one module can have adverse efforts on another. Integration testing is systematic testing for construction the program structure while at the same time conducting tests to uncover errors associated within the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole. Here correction is difficult because the isolation of cause is complicate by the vast expense of the entire program. Thus, in the integration testing stop, all the errors uncovered are corrected for the text testing steps.

6.2.3 System Testing

System testing is the stage of implementation that is aimed at ensuring that the system works accurately and efficiently for live operation commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct, then goal will be successfully achieved.

6.2.4 Acceptance Testing

This is a level of software testing where a system is tested for acceptability. The purpose of this test is to evaluate the system’s compliance with the business requirements and assess whether it is acceptable for delivery or not. The ability to reject invalid fingerprints or unrecognized fingerprints which are not in the database. This makes the applications stealth from cheating and misuse of the system.

Validation Criteria

* Numeric fields should be checked for non-numeric values. Similarly, text fields like names should not contain any numeric characters.
* All primary keys should be automatically generated to prevent the user from entering any existing key and on delete there should be cascading of primary key.
* Use of error and exceptional handling for each Create, Retrieve, Update, Delete and other important operations.
* In each form, no field which is not null able should be left blank.
* Input data should be validated before submission or posting, correct data formats should be submitted into the corresponding field inputs.
* In each form, no field which is not null should be left blank.

**6.2.5 Non-functional testing**

This involves testing how well the system meets the required software characteristics such as performance and usability. It tests the attribute of a system that do not relate to functionality. It is concerned with well a system works.

6.3 TEST CASES FOR MODULES

6.3.1 Test case for LCD and Motors.

6.3.1.1 Test case for LCD module

An LCD is tested on commands responding by display the direction given. The LCD hardware device is tested.

Table 6.1 LCD table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Title: Internet Of Things based Voice Controlled Wheelchair | | | | | |
| Test Case Template | | | | | |
| Test Case *#:* 1.1 | | | Test Designed by: prn-04 | | |
| Test Priority: Medium | | | Test Designed date: 11.07.2019 | | |
| Module Name: LCD | | | Test Executed by: prn-04 | | |
| Test Title: Verify LCD functionality | | | Test Executed date:11.07.2019 | | |
| Description: Test the prn-04lcd | | |  | | |
|  | | |  | | |
| Pre -conditions: LCD is given power | | | | | |
| Dependencies:5 Voltage power supply | | | | | |
|  | | | | | |
| Step | Test Steps | Test Data | Expected Result | Actual Result | Status |
| 1 | Connect the LCD |  | LCD displayed | As expected | Pass |
| 2 | Give a forward command | Forward command | LCD should display ‘Going forward’ | As expected | Pass |
| 3 | Give a left command | Left command | LCD should display ‘Going left’ | As expected | Pass |
| 4 | Give a right command | Right command | LCD should display ‘Going right’ | As expected | Pass |
| 5 | Give a back command | Back command | LCD should display ‘Going backward’ | As expected | Pass |
| 6 | Give a stop command | Stop command | LCD should display ‘Stopping’ | As expected | Pass |
| Post-conditions: LCD is able to display the direction due to given commands | | | | | |

6.3.1.2 Test case for Motors module

Table 6.2 Motors table

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case # | Test Scenario | Test Steps | Test Data | Expected Result | Actual Result | Status |
| 1 | Check response of motors when forward command. | 1.Power up the wheelchair.  2. Give the commands of the forward direction. | Forward command | Wheelchair should be able to move forward direction. | As expected | pass |
| 2 | Check the response of the motors when the left command is given | 1.Power up the wheelchair.  2.Give the left command. | Left command | Wheelchair should turn to the left direction. | As expected | pass |
| 3 | Check the response of the wheelchair when the right command is given | 1.Power up the wheelchair.  2.Give the right command | Right command | Wheelchair is expected to turn right when the right command is given | As expected | pass |
| 4 | Check the response of the wheelchair when the backward command is given | 1.Power up the wheelchair.  2.Give the backward command | Backward command | The wheelchair is expected to go backward when the backward command is given | As expected | pass |
| 5 | Checking the response of the wheelchair when the stop command is given. | 1.Power up the wheelchair.  2.Give the stop command | Stop command | The wheelchair is expected to stop when the stop command is given. | As expected | pass |

6.3.1.3 Test Case for Ultrasonic Sensor module

Table 6.3 Ultrasonic Sensor table

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case # | Test Scenario | Test Steps | Test Data | Expected Result | Actual Result | Status |
| 1 | Check whether the wheelchair stops when obstacle is In front of it | 1.Power up the wheelchair.  2. Give the command ‘go forward’ in the application  3.Add obstacle | Go forward | Wheelchair should stop | As expected | pass |
| 2 | Check whether the wheelchair will stop when obstacle is behind | 1.Power up the wheelchair.  2.Give the ‘go back’ command.  3.Add obstacle | Go back | Wheelchair should stop when it reaches the obstacle | As expected | pass |

6.3.1.4 Test case for Bluetooth transmission

Table 6.4 Bluetooth transmission table

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case # | Test Scenario | Test Steps | Test Data | Expected Result | Actual Result | Status |
| 1 | Check whether the forward string is being transmitted via bluetooth | 1.Power up the wheelchair.  2. Give the command ‘go forward’ in the application | Go forward | Arduino chip should receive ‘go forward’ string. | As expected | pass |
| 2 | Check whether the left string is being transmitted via bluetooth | 1.Power up the wheelchair.  2.Give the ‘go left’ command. | Go left | Arduino chip should receive ‘go forward’ string. | As expected | pass |
| 3 | Check whether the forward string is being transmitted via bluetooth | 1.Power up the wheelchair.  2.Give the right command | Right command | Wheelchair is expected to turn right when the right command is given | As expected | pass |
| 4 | Check whether the forward string is being transmitted via bluetooth | 1.Power up the wheelchair.  2.Give the backward command | Backward command | The wheelchair is expected to go backward when the backward command is given | As expected | pass |
| 5 | Check whether the forward string is being transmitted via bluetooth | 1.Power up the wheelchair.  2.Give the stop command | Stop command | The wheelchair is expected to stop when the stop command is given. | As expected | pass |

CHAPTER 7

CONCLUSION & FUTURE ENHANCEMENT

7.1 CONCLUSION

We are implementing an automatic wheelchair which has various advantages. It is operating in three different modes i.e. Voice commands, touch commands and by pressing buttons present on the wheelchair. Also, there are two sensors which increases the accuracy of the wheelchair. We did all of this for security purposes (To ensure the user won’t get any harm while using our product). The wheelchair will be economical and can be afforded to common people. The running cost of this system is much lower as compare to other systems used for the same purpose. The voice recognition system worked for most of the commands (over 95%). Only when a word is not properly vocalized, the system did not recognize it. The user can use manual system in case the voice command fails, there are buttons which indicate (forward, left, right &stop, Overall, users reported satisfaction with the system.

7.2 FUTURE ENHANCEMENT

The project has a very vast scope in future. The project can be implemented on intranet in future. Project can be updated in near future as and when requirement for the same arises, as it is very flexible in terms of expansion. With the proposed software Android application ready and fully functional the user is now able to use and hence move whole directions in a much better, accurate and error free manner. The following are the future scope for the project.

1. Smartphone will be able to be charged from the battery operating the wheelchair.
2. To be able to charge the battery by the motion of the wheelchair.
3. The ability to locate the wheelchair by using the GPS technology when it is connected to the internet connection.

APPENDIX

A1.SAMPLE CODES

Activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"  
xmlns:app="http://schemas.android.com/apk/res-auto"  
xmlns:tools="http://schemas.android.com/tools"  
android:id="@+id/navDrawer"  
android:layout\_width="match\_parent"  
android:layout\_height="match\_parent"  
android:fitsSystemWindows="true"  
tools:openDrawer="start">  
  
<include  
 layout="@layout/app\_bar\_main"  
android:layout\_width="match\_parent"  
android:layout\_height="match\_parent" />  
  
<android.support.design.widget.NavigationView  
android:id="@+id/nav\_view"  
android:layout\_width="wrap\_content"  
android:layout\_height="match\_parent"  
android:layout\_gravity="start"  
android:fitsSystemWindows="true"  
app:headerLayout="@layout/nav\_header\_main"  
app:menu="@menu/activity\_main\_drawer" />  
  
  
</android.support.v4.widget.DrawerLayout>

MainActivity.java

package giltwizy.hopedevelopers.com.iotwheelchair;  
  
import android.Manifest;  
import android.bluetooth.BluetoothAdapter;  
import android.content.ActivityNotFoundException;  
import android.content.DialogInterface;  
import android.content.Intent;  
import android.content.pm.PackageManager;  
import android.os.Bundle;  
import android.os.Handler;  
import android.os.Message;  
import android.os.Vibrator;  
import android.speech.RecognizerIntent;  
import android.support.annotation.NonNull;  
import android.support.design.widget.NavigationView;  
import android.support.v4.app.ActivityCompat;  
import android.support.v4.content.ContextCompat;  
import android.support.v4.view.GravityCompat;  
import android.support.v4.widget.DrawerLayout;  
import android.support.v7.app.ActionBarDrawerToggle;  
import android.support.v7.app.AlertDialog;  
import android.support.v7.app.AppCompatActivity;  
import android.support.v7.widget.Toolbar;  
import android.util.Log;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.view.View;  
import android.widget.TextView;  
import android.widget.Toast;  
import java.util.ArrayList;  
import java.util.Locale;  
  
  
public class Main extends AppCompatActivity implements NavigationView.OnNavigationItemSelectedListener{  
  
 public final String TAG = "Main";  
  
 private TextView status, resultTv;  
 private Bluetooth bt;  
 private Vibrator vibrator;  
  
 private final int recordAudio\_permission\_code = 2;  
 private static final int *voiceInputCode*= 11;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
super.onCreate(savedInstanceState);  
setContentView(R.layout.*activity\_main*);  
  
resultTv = findViewById(R.id.*voiceResult*);  
 status = findViewById(R.id.*textStatus*);  
  
 vibrator = (Vibrator)getSystemService(*VIBRATOR\_SERVICE*);  
  
findViewById(R.id.*connect*).setOnClickListener(new View.OnClickListener() {  
 public void onClick(View v) {  
connectService();  
 }  
 });  
  
findViewById(R.id.*disconnect*).setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
disconnectBluetooth();  
 }  
 });  
  
  
bt = new Bluetooth(this, mHandler);  
  
  
check\_recordAudio\_permission();  
  
  
 Toolbar toolbar = findViewById(R.id.*toolbar*);  
setSupportActionBar(toolbar);  
  
  
DrawerLayout drawer = findViewById(R.id.*navDrawer*);  
ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(  
 this, drawer, toolbar, R.string.*navigation\_drawer\_open*, R.string.*navigation\_drawer\_close*);  
drawer.addDrawerListener(toggle);  
toggle.syncState();  
  
NavigationViewnavigationView = findViewById(R.id.*nav\_view*);  
navigationView.setNavigationItemSelectedListener(this);  
  
 }  
  
  
 public void check\_recordAudio\_permission() {  
 //Check if Audio Recording permission is granted if not permission is requested  
 if (ContextCompat.*checkSelfPermission*(this,  
Manifest.permission.*RECORD\_AUDIO*) != PackageManager.*PERMISSION\_GRANTED*) {  
request\_recordAudio\_permission();  
 }  
 }  
  
  
 //method requesting Record Audio permission  
 private void request\_recordAudio\_permission() {  
 if (ActivityCompat.*shouldShowRequestPermissionRationale*(this,  
Manifest.permission.*RECORD\_AUDIO*)) {  
  
 new AlertDialog.Builder(this)  
 .setTitle("Permission needed")  
 .setMessage("This permission is needed so that this App can record the voice command")  
 .setPositiveButton("ok", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
ActivityCompat.*requestPermissions*(Main.this,  
 new String[]{Manifest.permission.*RECORD\_AUDIO*}, recordAudio\_permission\_code);  
 }  
 })  
 .setNegativeButton("cancel", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
dialog.dismiss();  
 }  
 })  
 .create().show();  
  
 } else {  
ActivityCompat.*requestPermissions*(this,  
 new String[]{Manifest.permission.*RECORD\_AUDIO*}, recordAudio\_permission\_code);  
 }  
 }  
  
  
 //Result of requesting the permission  
 @Override  
 public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {  
 if (requestCode == recordAudio\_permission\_code) {  
 if (grantResults.length> 0 &&grantResults[0] == PackageManager.*PERMISSION\_GRANTED*) {  
Toast.*makeText*(this, "Permission GRANTED", Toast.*LENGTH\_SHORT*).show();  
 } else {  
Toast.*makeText*(this, "Permission DENIED", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 }  
  
  
 public void getSpeechInput(View view) {  
  
 Intent intent = new Intent(RecognizerIntent.*ACTION\_RECOGNIZE\_SPEECH*);  
 //You can set here own local Language.  
intent.putExtra(RecognizerIntent.*EXTRA\_LANGUAGE\_MODEL*, RecognizerIntent.*LANGUAGE\_MODEL\_FREE\_FORM*);  
intent.putExtra(RecognizerIntent.*EXTRA\_LANGUAGE*, Locale.*getDefault*());  
intent.putExtra(RecognizerIntent.*EXTRA\_PROMPT*, "Hello, Tell me your direction");  
 try {  
startActivityForResult(intent, *voiceInputCode*);  
 } catch (ActivityNotFoundException a) {  
Log.*d*("voice", "Failed to get voice ");  
  
 }  
 }  
  
  
 //After getting the result of the string it is transfters the string command via bluetooth  
 @Override  
 protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
super.onActivityResult(requestCode, resultCode, data);  
  
 if (requestCode == *voiceInputCode*) {  
 if (resultCode == *RESULT\_OK* && null != data) {  
ArrayList<String> result = data.getStringArrayListExtra(RecognizerIntent.*EXTRA\_RESULTS*);  
resultTv.setText(result.get(0));  
 String recognizerResult = result.get(0).toLowerCase();  
bt.sendMessage(recognizerResult);  
Log.*d*("Transfer", "String transferred " + recognizerResult);  
 vibration();  
 }  
  
 }  
  
  
 }  
  
 private void vibration() {  
 if(vibrator.hasVibrator()){  
vibrator.vibrate(100);  
 }  
 }  
  
  
 public void connectService() {  
 try {  
status.setText(R.string.*connecting*);  
BluetoothAdapterbluetoothAdapter = BluetoothAdapter.*getDefaultAdapter*();  
 if (bluetoothAdapter.isEnabled()) {  
bt.start();  
bt.connectDevice("HC-05");  
Log.*d*(TAG, "Btservice started - listening");  
status.setText(R.string.*connected*);  
 } else {  
Log.*w*(TAG, "Btservice started - bluetooth is not enabled");  
status.setText(R.string.*notEnabled*);  
 }  
 } catch (Exception e) {  
Log.*e*(TAG, "Unable to start bt ", e);  
status.setText(R.string.*unableToConnect*);  
 }  
 }  
  
  
 private final Handler mHandler = new Handler() {  
 @Override  
 public void handleMessage(Message msg) {  
 switch (msg.what) {  
 case Bluetooth.*MESSAGE\_STATE\_CHANGE*:  
Log.*d*(TAG, "MESSAGE\_STATE\_CHANGE: " + msg.arg1);  
 break;  
 case Bluetooth.*MESSAGE\_WRITE*:  
Log.*d*(TAG, "MESSAGE\_WRITE ");  
 break;  
 case Bluetooth.*MESSAGE\_READ*:  
Log.*d*(TAG, "MESSAGE\_READ ");  
 break;  
 case Bluetooth.*MESSAGE\_DEVICE\_NAME*:  
Log.*d*(TAG, "MESSAGE\_DEVICE\_NAME " + msg);  
 break;  
 case Bluetooth.*MESSAGE\_TOAST*:  
Log.*d*(TAG, "MESSAGE\_TOAST " + msg);  
 break;  
 }  
 }  
 };  
  
 public void disconnectBluetooth() {  
BluetoothAdapterbluetoothAdapter = BluetoothAdapter.*getDefaultAdapter*();  
bluetoothAdapter.disable();  
Toast.*makeText*(getApplicationContext(), "Bluetooth Turned OFF", Toast.*LENGTH\_SHORT*).show();  
  
  
 }  
  
  
  
  
  
 @Override  
 public void onBackPressed() {  
DrawerLayout drawer = findViewById(R.id.*navDrawer*);  
 if (drawer.isDrawerOpen(GravityCompat.*START*)) {  
drawer.closeDrawer(GravityCompat.*START*);  
 } else {  
super.onBackPressed();  
 }  
 }  
  
  
 @Override  
 public booleanonCreateOptionsMenu(Menu menu) {  
 // Inflate the menu; this adds items to the action bar if it is present.  
getMenuInflater().inflate(R.menu.*main*, menu);  
 return true;  
 }  
  
  
 @Override  
 public booleanonOptionsItemSelected(MenuItem item) {  
 // Handle action bar item clicks here. The action bar will  
 // automatically handle clicks on the Home/Up button, so long  
 // as you specify a parent activity in AndroidManifest.xml.  
 int id = item.getItemId();  
  
 //noinspectionSimplifiableIfStatement  
// if (id == R.id.action\_settings) {  
// return true;  
// }  
  
 return super.onOptionsItemSelected(item);  
 }  
  
  
 @Override  
 public booleanonNavigationItemSelected(@NonNullMenuItem item) {  
 // Handle navigation view item clicks here.  
 int id = item.getItemId();  
  
 if (id == R.id.*voiceNav*) {  
openVoiceActivity();  
 } else if (id == R.id.*touchNav*) {  
openTouchActivity();  
  
 }  
  
DrawerLayout drawer = findViewById(R.id.*navDrawer*);  
drawer.closeDrawer(GravityCompat.*START*);  
 return true;  
 }  
  
  
 private void openVoiceActivity() {  
 Intent voiceIntent = new Intent(getApplicationContext(),Main.class);  
startActivity(voiceIntent);  
 overridePendingTransition(R.anim.*slide\_in\_right*,R.anim.*slide\_out\_left*);  
 }  
  
 private void openTouchActivity() {  
 Intent tigoIntent = new Intent(getApplicationContext(),Touch.class);  
startActivity(tigoIntent);  
 overridePendingTransition(R.anim.*slide\_in\_right*,R.anim.*slide\_out\_left*);  
 }  
  
  
}

ActivityTouch.xml

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayoutxmlns:android="http://schemas.android.com/apk/res/android"  
xmlns:app="http://schemas.android.com/apk/res-auto"  
xmlns:tools="http://schemas.android.com/tools"  
android:layout\_width="match\_parent"  
android:layout\_height="match\_parent"  
tools:context=".Touch">  
  
<LinearLayout  
android:id="@+id/firstRow"  
android:layout\_width="match\_parent"  
android:layout\_height="50dp"  
android:background="@color/colorPrimary"  
 style="?android:attr/buttonBarStyle"  
android:orientation="horizontal">  
  
<Button  
android:id="@+id/connect"  
android:layout\_width="0dp"  
android:layout\_height="match\_parent"  
android:layout\_weight="1"  
android:background="@drawable/button\_bg"  
android:shadowColor="@color/white"  
android:text="@string/connect"  
 style="?android:attr/buttonBarButtonStyle"  
android:textColor="@color/white" />  
  
<Button  
android:id="@+id/disconnect"  
android:layout\_width="0dp"  
android:layout\_height="match\_parent"  
android:layout\_weight="1"  
android:background="@drawable/button\_bg"  
android:text="@string/disconnect"  
 style="?android:attr/buttonBarButtonStyle"  
android:textColor="@color/white" />  
  
  
</LinearLayout>  
  
<LinearLayout  
android:id="@+id/secondRow"  
android:layout\_width="match\_parent"  
android:layout\_height="30dp"  
android:layout\_below="@id/firstRow"  
android:background="@color/colorPrimary"  
android:orientation="horizontal"  
android:weightSum="2">  
  
<TextView  
android:layout\_width="0dp"  
android:layout\_height="match\_parent"  
android:layout\_weight="1"  
android:text="@string/status"  
android:textAlignment="center"  
android:textColor="@color/white"  
android:textSize="18sp" />  
  
<TextView  
android:id="@+id/textStatus"  
android:layout\_width="16dp"  
android:layout\_height="match\_parent"  
android:layout\_weight="1"  
android:gravity="start"  
android:text="@string/disconnected"  
android:textAlignment="textStart"  
android:textColor="@color/white"  
android:textSize="18sp" />  
  
  
  
</LinearLayout>

A2. SCREENSHOTS

|  |  |
| --- | --- |
|  |  |
| Fig. A2.1 Home Screen | Fig. A2.2 Home Activity |

|  |  |
| --- | --- |
|  |  |
| Fig. A2.3 Navigation Bar | Fig. A2.4 Touch Activity |

|  |  |
| --- | --- |
|  |  |
| Fig. A2.5 Motor driver | Fig. A2.6 Ultrasonic sensor |
|  |  |
|  |  |
| Fig. A2.7 ATmega328p chip | Fig. A2.8 Liquid crystal display |

REFERENCES

BOOKS

1. Prof.r.S.Nipanikar,VinayGaikwad,ChetanChoudhari,RamGosavi,VishalHarne,2013,“Automaticwheelchair for physically disabled persons”.
2. Beginning C for Arduino, Second Edition: Learn C Programming for the Arduino, Jack Purdum
3. Learn to Program in Arduino C: 18 Lessons, from setup () to robots, William P. Osborne
4. Making Embedded Systems: Design Patterns for Great Software 1st Edition, Kindle Edition, Elecia White

https://www.karmanhealthcare.com/daily-problems-and-challenges-of-using-a-wheelchair/