

MOBILE CONTROLLED WHEELCHAIR WITH HEALTHCARE MONITORING FOR ELDERLY PEOPLE

Submitted in partial fulfillment of the requirements for the award of
Bachelor of engineering degree in Computer Science and Engineering
by

LUTUKURTHY VINDHYA MEGHANA (Reg.No. 37110411)

MADDURU SRUTHI REDDY (Reg. No. 37110415)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

**Accredited with Grade “A” by NAAC JEPPIAAR
NAGAR, RAJIV GANDHI SALAI,**

CHENNAI – 600 119

MARCH - 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this project report is the bonafide work of **LUTUKURTHY VINDHYA MEGHANA(Reg. No. 37110411)** and **MADDURU SRUTHI REDDY (Reg. No.37110415)** who carried out the project entitled "**MOBILE CONTROLLED WHEELCHAIR WITH HEALTHCARE MONITORING FOR ELDERLY PEOPLE**" under my supervision from **DECEMBER 2020** to **March 2021**.

Internal Guide

Mr. Albert Mayan J, M.E.,Ph.d.,

Head of the Department

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **LUTUKURTHY VINDHYA MEGHANA** hereby declare that the Project Report entitled **“MOBILE CONTROLLED WHEELCHAIR WITH HEALTHCARE MONITORING FOR ELDERLY PEOPLE”** is done by me under the guidance of **Mr. ALBERT MAYAN J,M.E., Ph.D.**, Department of Computer Science and Engineering at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala, M.E., Ph.D., Dean**, School of Computing, **Dr. S. Vigneswari, M.E., Ph.D., and Dr. L. Lakshmanan, M.E., Ph.D., Heads of the Department** of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Mr. ALBERT MAYAN J, M.E., Assistant Professor**, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Now a days there is an increase in the number of elderly and disabled people across the world. The elderly people need to get help from other people around to control the wheelchair and it creates a big problem when these people would like to travel alone. In order to assist physically challenged people, we developed a voice controlled wheelchair. The voice controlled wheelchair can provide easy access for persons who cannot control their movements. The user can control the wheelchair by voice commands. The proposed system aims at assisting people with physical disabilities and old aged people who are not able to move independently because of their weakness. The system allows the elderly people to live their life independently to an extent. Voice recognition technology is a new way of human interaction with machine. This project uses voice recognition for the controlling of wheelchair. It uses the voice recognition module which is interfaced with motors to take input commands from the user and moves the wheelchair.

TABLE OF CONTENTS

	ABSTRACT	v
	LIST OF FIGURES	viii
CHAPTER No.	TITLE	PAGE No.
1.	INTRODUCTION	1
	1.1 RESEARCH AND SIGNIFICANCE	1
	1.2 LITERATURE SURVEY	2
2.	AIM AND SCOPE	5
	2.1 AIM OF PROJECT	5
	2.1.1 OBJECTIVES	5
	2.2 SCOPES	5
	2.3 EXISTING SYSTEM	6
	2.3.1 DISADVANTAGES OF EXISTING SYSTEM	6
	2.4 PROPOSED SYSTEM	6
	2.4.1 ADVANTAGES OF PROPOSED SYSTEM	6
	2.5 SYSTEM REQUIREMENTS	7
3.	SYSTEM DESIGN & METHODOLOGY	8
	3.1 DESIGN	8
	3.1.1 CHARACTERISTICS	9
	3.1.2 APPLICATIONS	10
	3.2 POWER MANAGEMENT	11
	3.3 SENSORS	13
	3.4 METHODOLOGY	17
4.	SOFTWARE DEVELOPMENT METHODOLOGY	18
	4.1 GENERAL INTRODUCTION IOT	19
	4.1.1 OVERVIEW	20
	4.1.2 CHALLENGES	21

4.2	INTRODUCTION TO ARDUINO IDE	22
4.3	UBIDOTS	23
4.3.1	CREATING UBIDOTS	23
4.3.2	VARIABLES	23
4.3.3	EVENTS	24
5.	RESULTS AND DISCUSSION	26
6.	CONCLUSION AND FUTURE WORK	29
	REFERENCES	30
	APPENDIX	
	A. SOURCE CODE	31

LIST OF FIGURES

FIGURE No.	FIGURE NAME	PAGE No.
3.1	SYSTEM PROCESS	9
3.2	POWER SUPPLY	11
3.3	BLOCK DIAGRAM	13
3.4	HEARTBEAT DIODE	15
3.5	FLOWCHART OF VOICE	17
4.1	LAYERED ARCHITECTURE	21
4.2	ILLUSTRATION OF UBIDOTS	24
4.3	FLOWCHART OF WHEELCHAIR	25
5.1	WHEELCHAIR	26
5.2	APPLICATION	27
5.3	UBIDOTS	28
5.4	ALERT MAIL	28

CHAPTER 1

INTRODUCTION

INTRODUCTION

With the increasing birthrate and aging population, there have been many researches about electrical nursing robots, such as walking aid robots, muscle suit for nursing care, electric wheel chair, and so on. People with arms and hand impairment and elder people find it difficult to use a normal wheelchair as their hands are not capable of operating the normal wheelchair and cannot move it to any direction. Therefore, voice controlled wheel chair is built to overcome the problems faced by such people and enable them to operate the wheelchair. The wheelchair will be operated using the voice commands through the given input. Among them, electric wheel chair is widely used because it can help elderly people who have difficulty with driving wheel chair by their hand-power, and in usual, it does not need much help by care workers. Therefore, electric wheel chair has been developed for a long time. However, there are some problems on electric wheel chair. Since the radius of caster wheels which support wheel chair's stability are small, wheel chairs cannot go over a step even 3cm. Also, wheel chairs cannot move lateral direction in general, and they sometimes fall down and cause into serious accidents.

1.1 RESEARCH AND SIGNIFICANCE

To solve these problems, many researches have been carried out. For example, Wada proposed a Four-wheel-Drive (4WD) robotic platform for wheelchairs with medium-sized Omni-wheels and enabled to climb single step. R. Morales et al. designed wheelchair's mechanical system to climb up stairs by grasping a step and going over it. However, because of a numbers of sensors, their cost of failure and maintenance are high and they are difficult to use in daily lives. Therefore, a two-wheeled wheel chair is developed. The two-wheeled wheel chair was designed to realize "sensor less control". With few numbers of sensors, the two-wheeled wheel chair is superior in the view of cost reducing and high fault-tolerance, so it is suitable for physically handicapped people to use as useful mobility. The two-wheeled wheel

chair doesn't have any caster and move and stabilize with only two wheels. Its pitch angle of the body is measured by the gyro sensor, and the two in-wheel motors are controlled to stabilize the body. With the caster-less structure, the two-wheeled wheel chair has high mobility and go over a step easily. However, caster-less two-wheeled wheel chair system is unstable with pitch angle. Therefore, control is necessary to stabilize pitch angle. Although the unstable nature, with its high mobility, many stabilization theory for under actuated two-wheeled robot have been studied. For example, Sheelu Treesa Mathew et al. designed H_2 controller, Mahmood ul Hasan et al. used LQR controller. As for the two-wheeled wheel chair, Nakamura [6] realized the system stabilization by using the Lyapunov stabilizing theory, and enabled to move the two wheeled wheel chair forward and backward by the variation of user's center of mass. To improve the utility of the two wheeled wheel chair, many studies have done. Kuramats has implemented sensor less power assist of turning motion by introducing repulsive torque observer (RTOB)[8] to the two wheeled wheel chair. Kamatani has introduced joysticks to improve the operability of the two-wheeled wheel chair. Hirata has introduced backrest to improve the controllability by supporting human movement. However, there have been no research about lateral directional stability even though there are many reports about falling down accidents. One of the reasons of the accident is that wheel chairs cannot move lateral direction in general, so they are not be able to let a lateral force escape. In this paper, steering control method are proposed to solve this problem, and it enables two-wheeled wheel chair to move lateral direction. The validity of proposed method is checked through simulation compared with no steering control and changes of parameters.

1.2 LITERATURE SURVEY

Robotic Wheelchair Using Eye Blink Sensors and Accelerometer Provided with Home Appliance Control: Colleen Nelson, Nikitha S Badas, Saritha I G, Thejaswini S

According to a new report prepared jointly by theWorld Health Organizationand theWorld Bank, 15 percent of the world's population is disabled.The use of powered wheelchairs with high navigational intelligence is one of the great steps towards the integration of severely physically disabled and mentally handicapped people. Driving

a wheelchair is a tedious task for severely handicapped persons, unless they use the tongue to control the joystick. Simultaneously blind and paraplegic people deal with two problems, which creates uneasy situation for them, i.e locomotion and localization. Different systems are being developed to overcome the problems described above, allowing the end-user to perform safe movements and accomplish some daily life important tasks.

Sensor Based Eye Controlled Automated Wheelchair: Ms. Kale Taibai D., Ms. Patil Aishwarya S., Ms. Kamble Madhuri H., Prof. S. A. Jagtap

The project discuss about a wheel chair based on eye gesture and hand gestures. This project is help to those people who are not able doing any moment due to their illness, injury, or disability. They can move the wheel chair left or right byjust looking to the required direction, they can also start and stop the wheel chair, with other eye and hand gestures.

Robotic Wheelchair Controlled by Eye Blink andFace Orientation: Andreena Joseph1, D Antonio Thapeetha

Content: This paper describes and evaluates intelligent robotic wheelchair mainly for quadriplegic patients. Each year huge number of people suffers with a spinal cord injury and approximately half of these injuries result in quadriplegia. The loss of personal mobility is one of the major life changes brought on by quadriplegia, mobility becomes a lifelong struggle. Quadriplegics rely on power wheelchairs for mobility, but the hands-free controller systems currently available are obtrusive and expensive. The objective of this project was to design a power wheelchair with a novel control system for quadriplegics with Face Orientations and eye blink. USB camera and eye blink sensor was fixed in front of user's face. Face area was detected based on Ada Boost learning algorithm. Then facial landmarks were detected using Flandmark Detector.

Electric wheelchair control by gaze direction and blinking: Djoko Purwanto, Ronny Mardiyanto, Kohei Arai

We propose an electric wheelchair controlled by gaze direction and eye blinking. A camera is set up in front of a wheelchair user to capture image information. The sequential captured image is interpreted to obtain the gaze direction and eye blinking properties. The gaze direction is expressed by the horizontal angle of the gaze, and this is derived from the triangle formed by the centers of the eyes and the nose. The gaze direction and eye blinking are used to provide direction and timing commands, respectively. The direction command relates to the direction of movement of the electric wheelchair, and the timing command relates to the time when the wheelchair should move.

CHAPTER 2

AIM AND SCOPE

2.1 AIM

The main aim of the proposed system is that many elderly and disabled people depend on others in their daily especially in moving from one place to another on a wheelchair. The proposed system aims to make the elderly and disabled people independent to an extent. The system directs the wheelchair through voice.

2.1.1 OBJECTIVES

The main objective of voice controlled wheel chair system project is recommended to control a wheel chair by using speech recognition module. The system is designed to control a wheel chair using the voice of person. The objective of this project is to facilitate the movement of people who are disabled or handicapped and elderly people who are not able to move well. The goal of this system will allow certain people to live a life with less dependent on others for their movement as a daily need. Speech recognition technology is a key technology which will provide a new way of human interaction with machine or tools. Therefore the problems that they face can be solved by using speech recognition technology for the movement of wheel chair. This can be realized and optimized with use the smart phone device.

2.2 SCOPE

This model will help in building the strategies to make the elderly and disabled people more independent which will also be helpful for the society. This model helps us to keep a track on the elderly people's health conditions and monitor them closely.

2.3 EXISTING SYSTEM

Recently, the old person and the physically handicapped person who use a wheelchair are increasing. However, only two type wheelchairs by the hand operating and operating with the help of joystick, have come into wide use. The former type needs muscular strength for the operation and the latter type needs the skill or somebody else with them to move the wheelchair. Therefore, there is a problem that it is difficult for the old and the handicapped person to use these interfaces.

2.3.1DISADVANTAGES OF EXISTING SYSTEM:

- You may require assistance to get around.
- They require some strength to use.
- They can result in repetitive strain issues, especially in the shoulders.

2.4 PROPOSED SYSTEM

The proposed system uses voice commands to operate the wheelchair. To develop a system of improved facilities is the main aim of the proposed system. The proposed system can overcome the limitations of existing system. The proposed system will ensure that there is no need of any other human operator or care taker. The proposed system helps wheelchair to communicate with each other. This communication is used to know whether the task assigned to wheelchair is completed or not. Here we also monitor the health condition of the elderly person. When the person shows any signs of abnormal condition, an alert message is sent to the respective people like doctors, relatives.

2.4.1 ADVANTAGES OF PROPOSED SYSTEM

- Reduced man power
- User friendly
- Person can become more independent

2.5 SYSTEM REQUIREMENTS

Hardware required:

- NodeMCU
- Heartbeat sensor
- Pressure sensor
- DHT11 sensor
- Motor driver
- Motor

Software required:

- Embedded C
- Arduino IDE

CHAPTER 3

SYSTEM DESIGN & METHODOLOGY

3.1 DESIGN

EMBEDDED SYSTEM

An embedded system is a microprocessor-based computer hardware system with software that is designed to perform a dedicated function, either as an independent system or as a part of a large system. At the core is an integrated circuit designed to carry out computation for real-time operations. In contrast, a general-purpose computer, such as a personal computer, can do many different tasks depending on programming. Embedded systems have become very important today as they control many of the common devices we use.

Since the embedded system is dedicated to specific tasks, design engineers can optimize it, reducing the size and cost of the product, or increasing the reliability and performance. Some embedded systems are mass-produced, benefiting from economies of scale.

Physically, embedded systems range from portable devices such as digital watches and MP3 players, to large stationary installations like traffic lights, factory controllers, or the systems controlling nuclear power plants. Complexity varies from low, with a single microcontroller chip, to very high with multiple units, peripherals and networks mounted inside a large chassis or enclosure.

In general, "embedded system" is not an exactly defined term, as many systems have some element of programmability. For example, Handheld computers share some elements with embedded systems — such as the operating systems and microprocessors which power them — but are not truly embedded systems, because they allow different applications to be loaded and peripherals to be connected.

Embedded systems provide several functions :

- Monitor the environment; embedded systems read data from input sensors. This data is then processed and the results displayed in some format to a user or users
- Control the environment; embedded systems generate and transmit commands for actuators.
- Transform the information; embedded systems transform the data collected in some meaningful way, such as data compression/decompression

Although interaction with the external world via sensors and actuators is an important aspect of embedded systems, these systems also provide functionality specific to their applications. Embedded systems typically execute applications such as control laws, finite state machines, and signal processing algorithms. These systems must also detect and react to faults in both the internal computing environment as well as the surrounding electromechanical systems.

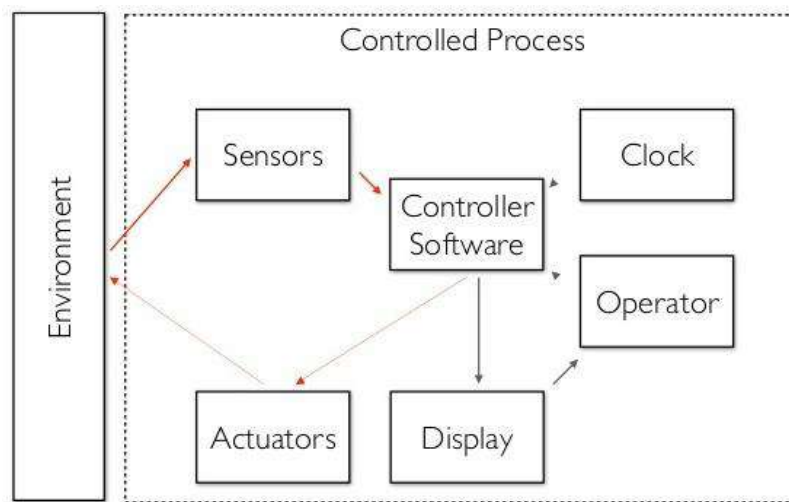


FIG 3.1: PROCESS

3.1.1 Characteristics of embedded systems

Embedded systems are characterized by a unique set of characteristics. Each of these characteristics imposed a specific set of design constraints on embedded systems designers. The challenge to designing embedded systems is to conform to the specific set of constraints for the application.

3.1.2 Application Specific Systems

Embedded systems are not general-purpose computers. Embedded system designs are optimized for a specific application. Many of the job characteristics are known before the hardware is designed. This allows the designer to focus on the specific design constraints of a well-defined application. As such, there is limited user reprogram ability. Some embedded systems, however, require the flexibility of reprogram ability. Programmable DSPs are common for such applications.

Harsh environment:

Many embedded systems do not operate in a controlled environment. Excessive heat is often a problem, especially in applications involving combustion (e.g., many transportation applications). Additional problems can be caused for embedded computing by a need for protection from vibration, shock, lightning, power supply fluctuations, water, corrosion, fire, and general physical abuse.

System safety and reliability:

As embedded system complexity and computing power continue to grow, they are starting to control more and more of the safety aspects of the overall system. These safety measures may be in the form of software as well as hardware control. Mechanical safety backups are normally activated when the computer system loses control in order to safely shut down system operation. Software safety and reliability is a bigger issue. Software doesn't normally "break" in the sense of hardware. However software may be so complex that a set of unexpected circumstances can cause software failures leading to unsafe situations. Discussion of this topic is outside the scope of this book, but the challenges for embedded designers include designing reliable software and building cheap, available systems using unreliable components. The main challenge for embedded system designers is to obtain low-cost reliability with minimal redundancy.

Small and low weight:

Many embedded computers are physically located within some larger system. The form factor for the embedded system may be dictated by aesthetics. For example, the form factor for a missile may have to fit inside the nose of the missile. One of the challenges for embedded systems designers is to develop non-rectangular geometries for certain solutions. Weight can also be a critical constraint. Embedded

automobile control systems, for example, must be light weight for fuel economy. Portable CD players must be light weight for portability purposes.

Cost sensitivity:

Cost is an issue in most systems, but the sensitivity to cost changes can vary dramatically in embedded systems. This is mainly due to the effect of computer costs have on profitability and is more a function of the proportion of cost changes compared to the total system cost.

3.2 POWER MANAGEMENT:

Embedded systems have strict constraints on power. Given the portability requirements of many embedded systems, the need to conserve power is important to maintain battery life as long as possible. Minimization of heat production is another obvious concern for embedded systems. All electronic circuits works only in low DC voltage, so we need a power supply unit to provide the appropriate voltage supply for their proper functioning .This unit consists of transformer, rectifier, filter & regulator. AC voltage of typically 230volts rms is connected to a transformer voltage down to the level to the desired ac voltage. A diode rectifier that provides the full wave rectified voltage that is initially filtered by a simple capacitor filter to produce a dc voltage. This resulting dc voltage usually has some ripple or ac voltage variation . A regulator circuit can use this dc input to provide dc voltage that not only has much less ripple voltage but also remains the same dc value even the dc voltage varies some what, or the load connected to the output dc voltages changes.

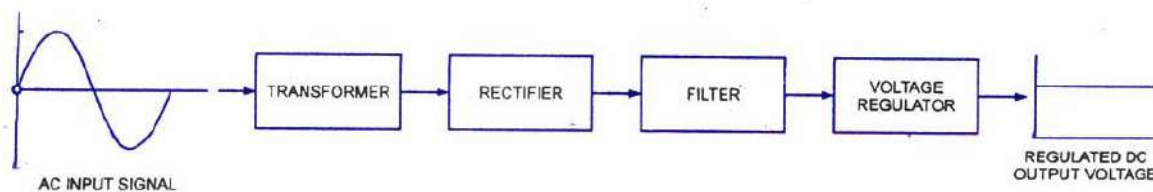


Fig 3.2.General Block Of Power Supply Unit

TRANSFORMER:

A transformer is a static piece of which electric power in one circuit is transformed into electric power of same frequency in another circuit. It can raise or lower the voltage in the circuit, but with a corresponding decrease or increase in current. It works with the principle of mutual induction. In our project we are using a step down transformer to providing a necessary supply for the electronic circuits. Here we step down a 230volts ac into 12volts ac.

RECTIFIER:

A dc level obtained from a sinusoidal input can be improved 100% using a process called full wave rectification. Here in our project for full wave rectification we use bridge rectifier. From the basic bridge configuration we see that two diodes(say D2 & D3) are conducting while the other two diodes (D1 & D4) are in off state during the period $t = 0$ to $T/2$. Accordingly for the negative cycle of the input the conducting diodes are D1 & D4 .Thus the polarity across the load is the same.

In the bridge rectifier the diodes may be of variable types like 1N4001, 1N4003, 1N4004, 1N4005, 1N4007 etc.... can be used. But here we use 1N4007, because it can withstand up to 1000v.

FILTERS:

In order to obtain a dc voltage of 0 Hz, we have to use a low pass filter. So that a capacitive filter circuit is used where a capacitor is connected at the rectifier output & a dc is obtained across it. The filtered waveform is essentially a dc voltage with negligible ripples & it is ultimately fed to the load.

REGISTERS:

The controller IC has two 8 bit registers, an instruction register (IR) and a data register (DR). The IR stores the instruction codes and address information for display data RAM (DD RAM) and character generator RAM (CG RAM). The IR can be written, but not read by the MPU.

When an address code is written to IR, the data is automatically transferred from the DD RAM or CG RAM to the DR. data transfer between the MPU is then completed when the MPU reads the DR. likewise, for the next MPU read of the DR, data in DD RAM or CG RAM at the address is sent to the DR automatically. Similarly, for the MPU write of the DR, the next DD RAM or CG RAM address is selected for the write operation.

REGULATORS:

The output voltage from the capacitor is more filtered & finally regulated. The voltage regulator is a device, which maintains the output voltage constant irrespective of the change in supply variations, load variations & temperature changes. Here we use fixed voltage regulator namely LM7805. The IC LM7805 is a +5v regulator which is used for microcontroller.

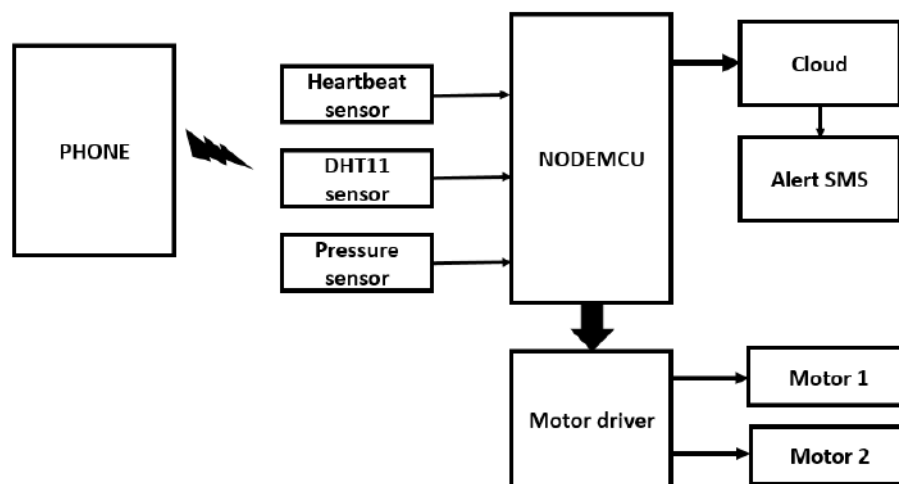


FIG 3.3:Block diagram

3.3 SENSORS

HEARTBEAT SENSOR

A person's heartbeat is the sound of the valves in his/hers heart contracting or expanding as they force blood from one region to another. The number of times the

heart beats per minute (BPM), is the heart beat rate and the beat of the heart that can be felt in any artery that lies close to the skin is the pulse.

Principle of Heartbeat Sensor

The principle of heart beat sensor, when the finger tissue is illuminated using a light source, the light is transmitted after getting modulated i.e. a part getting absorbed by the blood and the rest being transmitted. This modulated light is received by the light detector.

Here a Light Dependent Resistor (LDR) is used as a light detector. It works on the principle that when light falls on the resistor, its resistance changes. As the light intensity increases, the resistance decreases. Thus the voltage drop across the resistor decreases.

Here a comparator is used which compares the output voltage from the LDR to that of the threshold voltage. The threshold voltage is the voltage drop across the LDR when the light with fixed intensity, from the light source falls directly on it. The inverting terminal of the comparator LM358 is connected to the potential divider arrangement which is set to the threshold voltage and the non-inverting terminal is connected to the LDR. When a human tissue is illuminated using the light source, the intensity of the light reduces. As this reduced light intensity falls on the LDR, the resistance increases and as a result the voltage drop increases. When the voltage drop across the LDR or the non-inverting input exceeds that of the inverting input, a logic high signal is developed at the output of the comparator and in case voltage drop being lesser a logic low output is developed. Thus the output is a series of pulses. These pulses can be fed to the Microcontroller which accordingly processes the information to get the heart beat rate and this is displayed on the Display interfaced to the Microcontroller.

The basic heartbeat sensor consists of a light emitting diode and a detector like a light detecting resistor or a photodiode. The heart beat pulses causes a variation in the flow of blood to different regions of the body. When a tissue is illuminated with the light source, i.e. light emitted by the led, it either reflects (a finger tissue) or

transmits the light. Some of the light is absorbed by the blood and the transmitted or the reflected light is received by the light detector. The amount of light absorbed depends on the blood volume in that tissue. The detector output is in form of electrical signal and is proportional to the heart beat rate.

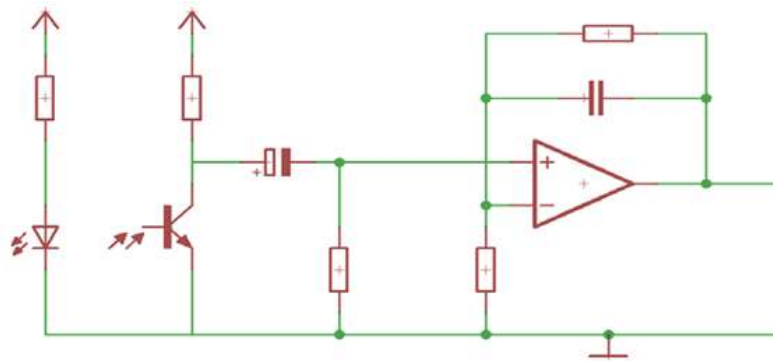


FIG 3.4 HEARTBEAT SENSOR DIODE

To achieve the task of getting the AC signal, the output from the detector is first filtered using a 2 stage HP-LP circuit and is then converted to digital pulses using a comparator circuit or using simple ADC. The digital pulses are given to a microcontroller for calculating the heart beat rate, given by the formula-BPM(Beats per minute) = $60 * f$ (f-frequency)

DHT11 SENSOR

The DHT11 sensor can also be used to sense the body temperature. The DHT11 sensor comes in a single row 4-pin package and operates from 3.5 to 5.5V power supply. It can measure temperature from 0-50 °C with an accuracy of $\pm 2^{\circ}\text{C}$ and ranging from 20-95% with an accuracy of $\pm 5\%$. The sensor provides fully calibrated digital outputs for the two measurements. It has got its own proprietary 1-wire protocol, and therefore, the communication between the sensor and a microcontroller is not possible through a direct interface with any of its peripherals. The protocol must be implemented in the firmware of the MCU with precise timing required by the sensor. The following timing diagrams describe the data transfer protocol between a MCU and the DHT11 sensor. The MCU initiates data transmission by issuing a *Start* signal. The MCU pin must be configured as output for this purpose. The MCU first pulls the data line low for at least 18 ms and then pulls it

high for next 20-40 s before it releases it. Next, the sensor responds to the MCU *Start* signal by pulling the line low for 80s followed by a logic high signal that also lasts for 80s. Remember that the MCU pin must be configured to input after finishing the *Start* signal. Once detecting the response signal from the sensor, the MCU should be ready to receive data from the sensor. The sensor then sends 40 bits (5 bytes) of data continuously in the data line.

PRESSURE SENSOR

A pressure sensor measures pressure, typically of gases or liquids. Pressure is an expression of the force required to stop a fluid from expanding, and is usually stated in terms of force per unit area. A pressure sensor usually acts as a transducer; it generates a signal as a function of the pressure imposed. For the purposes of this article, such a signal is electrical.

Pressure sensors are used for control and monitoring in thousands of everyday applications. Pressure sensors can also be used to indirectly measure other variables such as fluid/gas flow, speed, water level, and altitude. Pressure sensors can alternatively be called pressure transducers, pressure transmitters, pressure senders, pressure indicators, piezometers and manometers, among other names.

Pressure sensors can vary drastically in technology, design, performance, application suitability and cost. A conservative estimate would be that there may be over 50 technologies and at least 300 companies making pressure sensors worldwide.

There is also a category of pressure sensors that are designed to measure in a dynamic mode for capturing very high speed changes in pressure. Example applications for this type of sensor would be in the measuring of combustion pressure in an engine cylinder or in a gas turbine. These sensors are commonly manufactured out of piezoelectric materials such as quartz.

Some pressure sensors, such as those found in some traffic enforcement cameras, function in a binary (off/on) manner, i.e., when pressure is applied to a pressure sensor, the sensor acts to complete or break an electrical circuit. These types of sensors are also known as a pressure switch.

3.4 METHODOLOGY

Voice recognition technology is a new way of human interaction with machine. Voice recognition systems enable consumers to interact with technology simply by speaking to it, enabling hands-free requests, reminders and other simple tasks.

Voice recognition software on computers requires that analog audio be converted into digital signals, known as analog-to-digital conversion. For a computer to decipher a signal, it must have a digital database, or vocabulary, of words or syllables, as well as a speedy means for comparing this data to signals. The speech patterns are stored on the hard drive and loaded into memory when the program is run. A comparator checks these stored patterns against the output of the A/D converter -- an action called pattern recognition.

In practice, the size of a voice recognition program's effective vocabulary is directly related to the random access memory capacity of the computer in which it is installed. A voice recognition program runs many times faster if the entire vocabulary can be loaded into RAM, as compared with searching the hard drive for some of the matches. Processing speed is critical, as well, because it affects how fast the computer can search the RAM for matches.

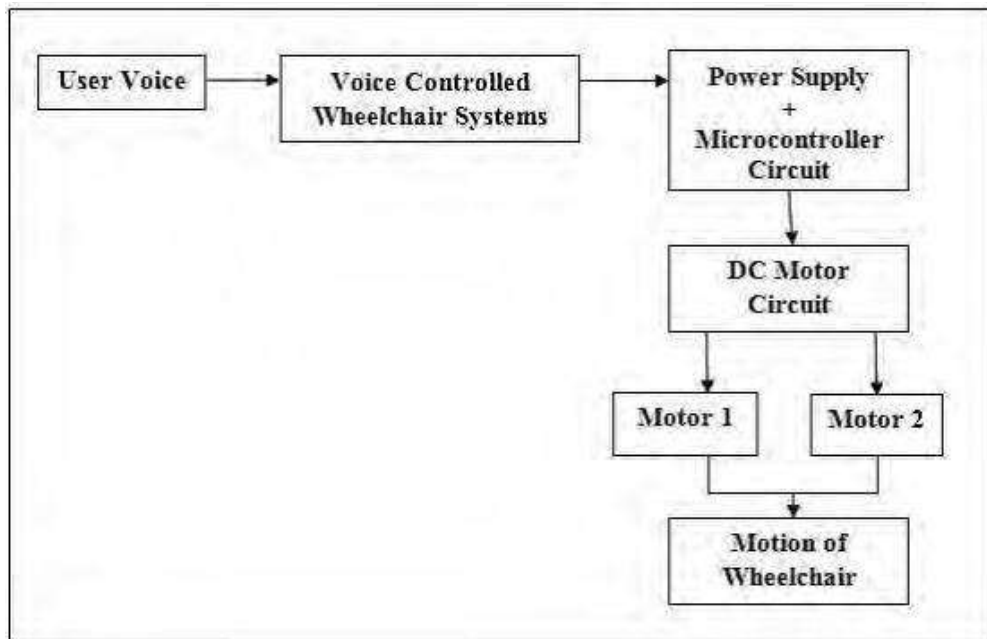


FIG 3.5: FLOWCHART OF VOICE CONTROL

The uses for voice recognition have grown quickly as AI, machine learning and consumer acceptance have matured. In-home digital assistants from Google to Amazon to Apple have all implemented voice recognition software to interact with users. The way consumers use voice recognition technology varies depending on the product, but it can include transcribing voice to text, setting up reminders, searching the internet, and responding to simple questions and requests, such as playing music or sharing weather or traffic information. Voice recognition enables consumers to multitask by speaking directly to their Google Home, Amazon Alexa or other voice recognition technology. By using machine learning and sophisticated algorithms, voice recognition technology can quickly turn your spoken work into written text.

While accuracy rates are improving, all voice recognition systems and programs make errors. Background noise can produce false input, which can be avoided by using the system in a quiet room. There is also a problem with words that sound alike, but that are spelled differently and have different meanings -- for example, hear and here. This problem might someday be largely overcome using stored contextual information. However, this will require more RAM and faster processors than are currently available in personal computers.

CHAPTER 4

SOFTWARE DEVELOPMENT METHODOLOGY

4.1 GENERAL INTRODUCTION OF INTERNET OF THINGS (IOT)

The Future Internet goal is to provide an infrastructure to have an immediate access to information about the physical world and its objects. Physical objects can be applicable to different application domains, such as e-health, warehouse management, etc. Each application domain may have different types of physical devices. Each physical device can have its own specifications, which is required to use in order to interact with it. To achieve the future Internet goal, a layered vision is required that can facilitate data access. Internet of Things (IoT) is a vision that aims to integrate the virtual world of information to the real world of devices through a layered architecture.

The term Internet of Things consists of two words, namely *Internet* and *Things*. *Internet* refers to the global network infrastructure with scalable, configurable capabilities based on interoperable and standard communication protocols. Things are physical objects or devices, or virtual objects, devices or information, which have identities, physical attributes and virtual personalities, and use intelligent interfaces. For instance, a virtual object can represent an abstract unit of sensor nodes that contains metadata to identify and discover its corresponding sensor nodes. Therefore, IoT refers to the *things* that can provide information from the physical environment through the Internet.

Middleware is as an interface between the hardware layer and the application layer, which is responsible for interacting with devices and information management. The role of a middleware is to present a unified programming model to interact with devices. A middleware is in charge of masking the heterogeneity and distribution problems that we face when interacting with devices.

4.1.1 OVERVIEW

Internet of Things (IoT) has increasingly gained attention in industry to interact with different types of devices. IoT can have influence on industry and society by integrating physical devices into information networks. IoT impacts can be on different perspectives, namely for private and business users. From the perspective of a private user, IoT has effect on both working and personal fields, such as smart homes and offices, e-health and assisted living. From the aspect of a business user the impacts would be in fields such as automation and industrial manufacturing, logistics, business process management, intelligent transportation of people and goods.

IoT integrates physical things into information networks. IoT covers the overall infrastructure, including software, hardware and services, which is used to support these information networks. The integrated physical things can exchange data about the physical properties and information that they sense in their environment. To identify devices, we can use identification technologies like for example RFID, which allow each device be uniquely identified.

IoT has a layered architecture designed to answer the demands of various industries, enterprises and society shows a generic layered architecture for IoT that consist of five layers, which are discussed, in the following:

- Edge Technology layer :

This is a hardware layer that consists of embedded systems, RFID tags, sensor networks and all of the other sensors in different forms. This hardware layer can perform several functions, such as collecting information from a system or an environment, processing information and supporting communication.

- Access Gateway layer :

This layer is concerned with data handling, and is responsible for publishing and subscribing the services that are provided by the *Things*, message routing, and hovelling the communication between platforms.

- **Middleware layer :**

This layer has some critical functionality, such as aggregating and filtering the received data from the hardware devices, performing information discovery and providing access control to the devices for applications.

- **Application layer :**

This layer is responsible for delivering various application services. These services are provided through the middleware layer to different applications and users in IoT-based systems. The application services can be used in different industries such as, logistics, retail, healthcare, etc.

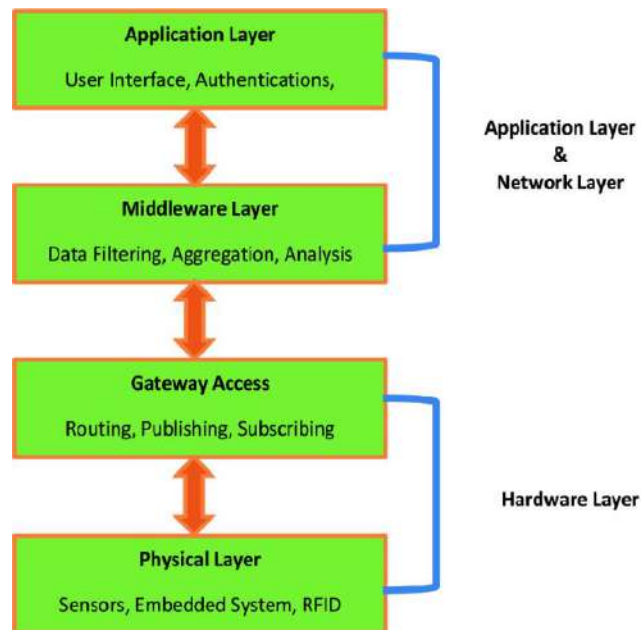


Fig4.1 Layered architecture of the Internet of Things (IoT)

4.1.2 CHALLENGES

IoT-based systems can be used for different purposes and areas, so that, we have to face different challenges. In this section, we explain some of the challenges that need to be considered in research activities:

Edge technology:

At the hardware level, more research efforts are required to develop the technology of embedded devices, sensors, actuators and (passive and active) identification, since an IoT-based system must be able to gather sufficient information about the real world by employing a large variety of devices and environments. Thus, there more work is still required to connect heterogeneous devices and deploy them in IoT applications, and to provide support for new devices.

Networking technologies :

In IoT, *things* are connected through different kinds of networks, i.e. mobile, wired and wireless network. These networks support bi-directional communication at different levels among the real world objects, applications and services that are employed by the IoT applications. This highly distributed structure should provide interconnection with low energy consumption, while distributed data can cause privacy issues.

Middleware system :

In IoT, we have heterogeneous devices and networks. Their heterogeneity can potentially increase with new technologies. To facilitate the use of these devices by IoT applications, we should shield their heterogeneity. Therefore, we need to develop a secure, scalable and semantically enriched service-oriented middleware to cope with the heterogeneity of devices.

Platform services :

They support a superior management of all involved devices in an integrated way, ensuring scalability, high availability, and the safe and secure execution of the requested functionality from devices.

4.2 INTRODUCTION TO ARDUINO IDE

The Arduino IDE (Integrated Development Environment) is used to write the computer code and upload this code to the physical board. The arduino IDE is very simple and this simplicity is probably one of the main reason arduino became so popular. We can certainly

state that being compatible with the arduino IDE is now one of the main requirements for a new microcontroller board. Over the years, many useful features have been added to the arduino IDE and can now be managed by third-party libraries and boards from the IDE, and still keep the simplicity of programming the board. The arduino board is connected to a computer via USB, where it connects with the IDE. The user writes the arduino code in the IDE, and then uploads it to the microcontroller which executes the code, interacting with inputs and outputs such as sensors, motors and lights.

4.3 UBIDOTS

A Ubidots' device is a virtual representation of a data-source or simply, an asset taking sensor data and transmitting said data through a connection protocol to Ubidots' cloud.

4.3.1 Creating UBIDOTS:

- Devices are created automatically in Ubidots the first time a dot is received to a user's private TOKEN or an Organizational TOKEN.
- Clicking the "+" icon in the top right corner of any single device's screen.
- Creating a Device Type which then creates a new device with pre-determined device properties, appearance, and variables and automate the on-boarding of thousands of devices.

4.3.2 Variables:

Once a device is created and receiving data from your hardware or another 3rd party data-source, the data will be presented in its raw or calculated form as a variable.

Types of Variables:

- Default - raw data coming from devices (people counted).
- Synthetic - correspond to statistical or arithmetical operations of default variables in a determined time-frame (e.g. average daily traffic this month).

Creating variables: There are two ways to create variables.

1. Assigning a label as a variable within your hardware's code.
2. Clicking the "+" icon found in any single device screen and assign a name which will also correspond to the variable's label. By default, Ubidots assigns an additional unique identifier to the variable called Variable ID. To find both the variable's ID and variable's label, select the "i" icon from the variables card.

4.3.3 Events

In Ubidots, Events are messages triggered and delivered through Email, SMS, Telegrams, Slack, Voice Call or web messages based on a customized design rule created in the application.

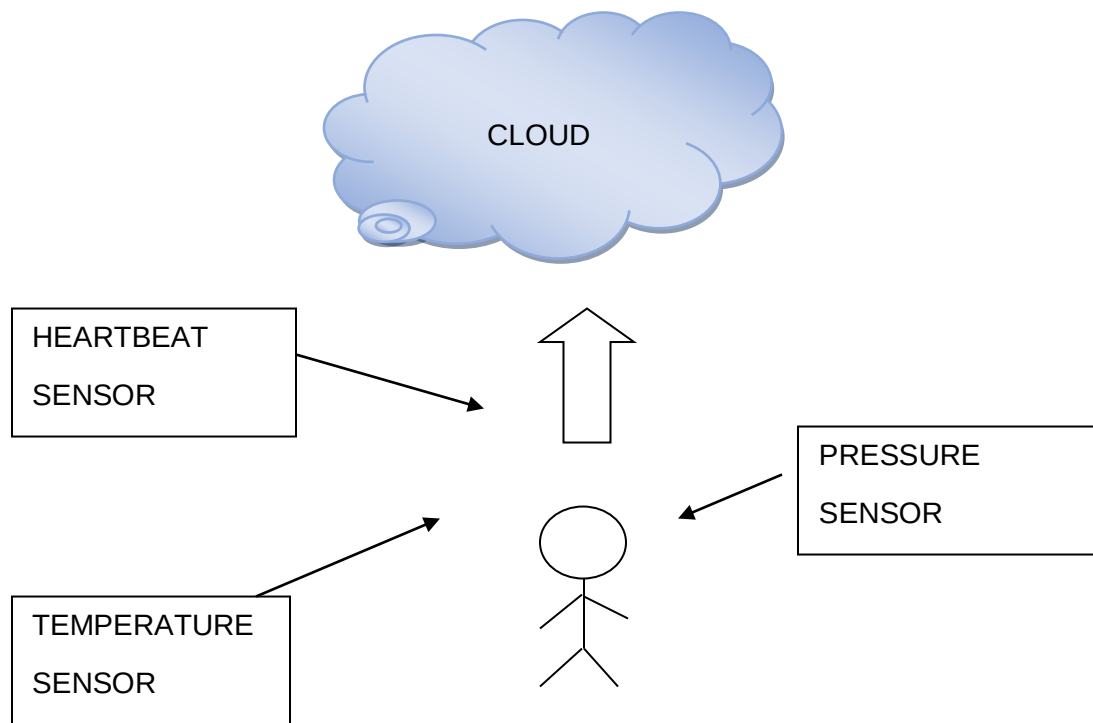


FIG 4.2 ILLUSTRATION OF UBIDOTS

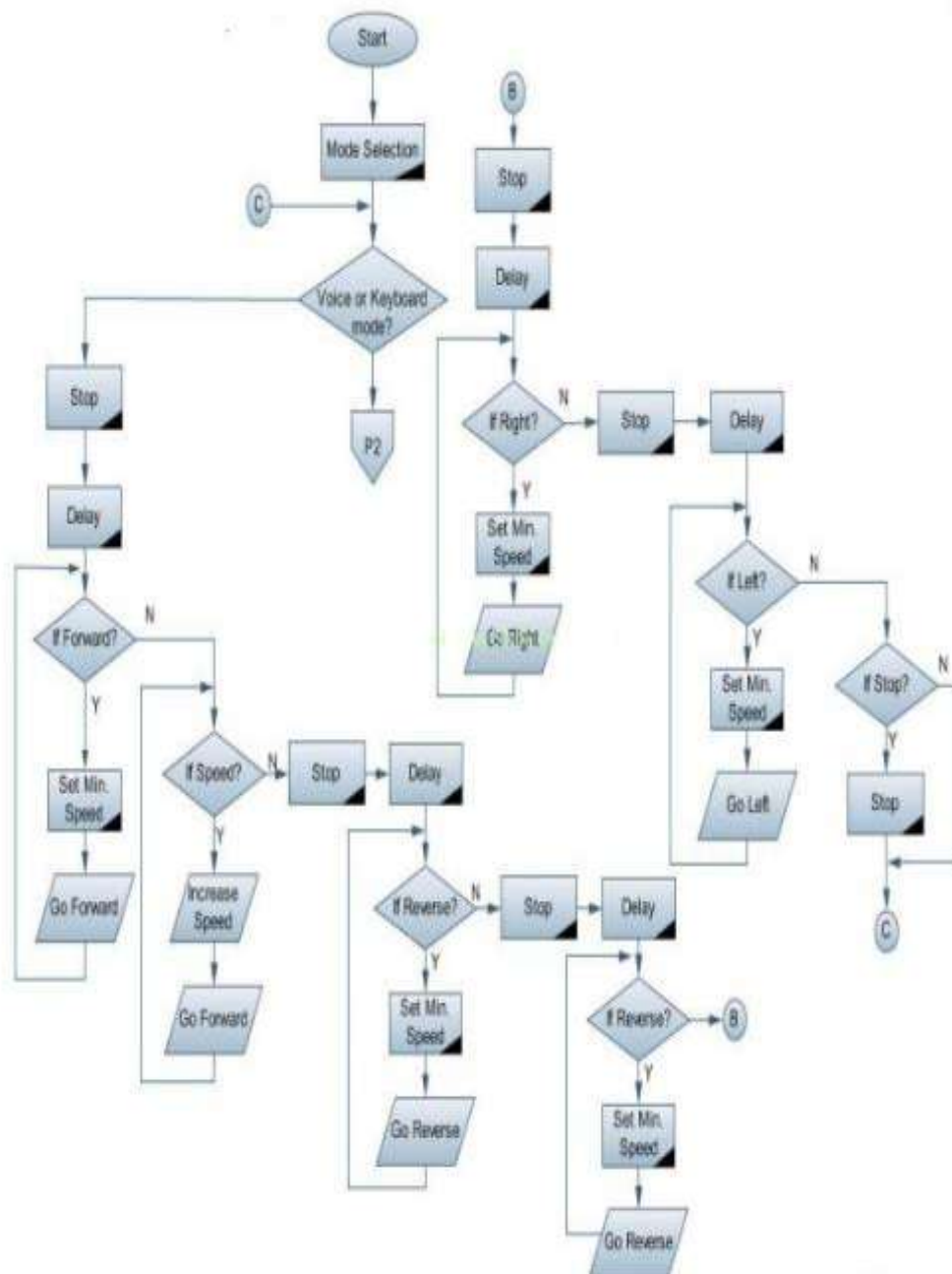


FIG 4.3 FLOWCHART OF WHEELCHAIR

CHAPTER 5

RESULT AND DISCUSSION

5.1 RESULTS

Here we created a mobile controlled wheelchair robot which consists of a 12V battery, nodemcu to which all the sensors such as the heartbeat sensor, dht11 sensor, pressure sensor are connected with wires.

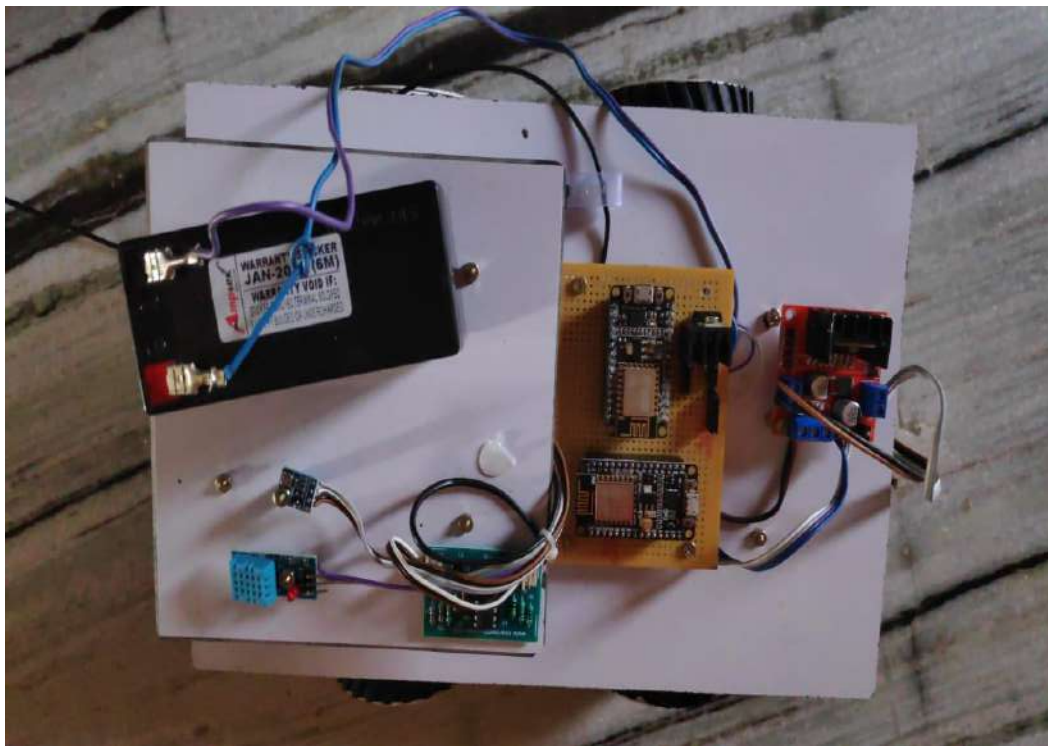


FIG 5.1 WHEELCHAIR

The arduino is controlled by a wifi controlled application which should be downloaded on the person's mobile. When the user gives instruction as forward the wheelchair robot moves forward and same for backward, left, right and stop.

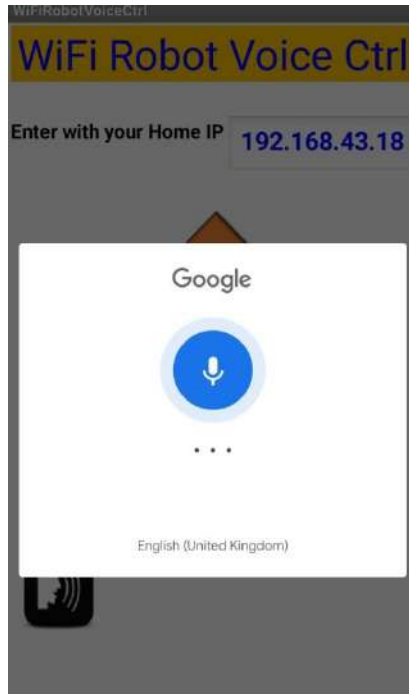


FIG 5.2 APPLICATION

The health conditions of the person can also be monitored using the wheelchair. We have a clip like thing attached to the wheelchair which monitors the pulse and pressure of the person. The health data is stored in a cloud platform called ubidots. The ubidots shows the data when logged in. And if the temperature or pulse exceeds it also sends an alert mail to the saved id.

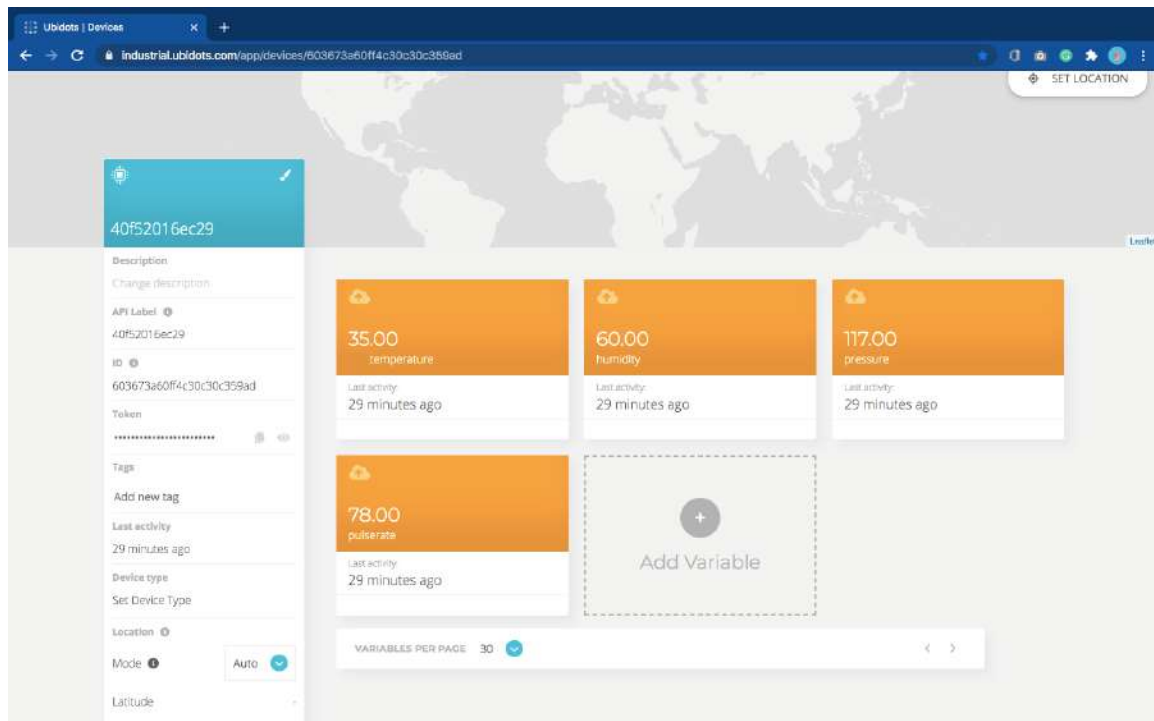


FIG 5.3 DATA IN UBIDOTS

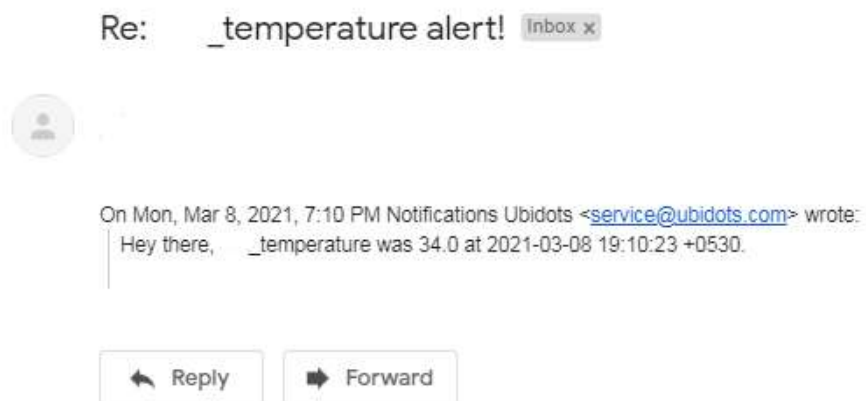


FIG 5.4 ALERT MAIL

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

In this proposed work we have created a wheel chair which is controlled by voice commands. The design and implementation of a voice controlled wheelchair for disabled people using arduino and voice recognition module for controlling the motion of a wheelchair is designed. The direction of the wheelchair now can be selected using the specified voice commands. The design not only reduce the manufacture cost compared with present market but also will give great competitive with other types of electrical wheelchair. The only thing needed to ride the wheelchair is the synthetic voice commands of the person. A system that can directly enhanced the lifestyle of a physically disabled person in the community is implemented. This project has many advantages like safety, comfort, energy saving, full automation etc. We can also monitor the elderly person's health condition and an alert message is sent if they display any abnormal conditions.

6.2 FUTURE WORK

Voice recognition module is used to develop the voice recognition system. Voice recognition issues a Command to control the movement of wheelchair. For not to occur disorder during recognize the user voice, this system works in a quiet environment. Furthermore, the pronunciations accuracy must be ensured and the word-related (voice) the users voice must clear in short distance on microphone was essential in this innovation. Using gear box we can produce high speed moving wheelchair. PWM modulation can also increase speed. Solar Panel can also be used to charge the battery for power supply to the components required to drive the wheelchair. The wheelchair can also include the gesture feature to operate the wheelchair. Wheelchair only can function properly when the weight of the load for this system must be below 50 kilogram. Obstacle avoidance sensors are used.

6.3 REFERENCES

- [1] A. Iqbal, M. Arifin, and A. Hossain, "Smart Home Appliance Control System for Physically Disabled People using Kinect and X10", in 2016 5th International Conference on Informatics, Electronics and Vision (ICIEV), pp. 891-896.
- [2] C. Z. Yue, and S. Ping. "Voice Activated Smart Home Design and Implementation" in 2017 2nd International Conference on Frontiers of Sensors Technology, pp. 489-492.
- [3] D. H. Stafanov, Z. Bien, and W. Bang, "The Smart House for Older Persons and Persons with Physical Disabilities: Structure, Technology Arrangements, and Perspectives. IEEE Transaction on Neural Systems and Rehabilitation Engineering, vol, 12, no. 2, pp. 228-250, June 2004.
- [4] D. Sunehra, and V. Tejaswi, "Implementation of Speech Based Home Automation System using Bluetooth and GSM" in International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)-2016, pp. 807-813.
- [5] H. Basanta, Y. Huang, and T. Lee, "Assistive Design for Elderly Living Ambient using Voice and Gesture Recognition System" in 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff Center, Banff, Canada.
- [6] J. Waleed, A. M. abduldaim, T. M. Hasan, and Q. S. Mohaisin, "Smart Home as a New Trend, a Simplicity Led to Revolution" in 2018 1st International Scientific Conference of Engineering Sciences – 3rd Scientific Conference of Engineering Sciences (ISCES), pp. 30-33.

APPENDIX

SOURCE CODE

```
#include <Adafruit_Sensor.h>
#include <DHT.h>
#include <DHT_U.h>

#define DHTPIN D4

#define DHTTYPE DHT11

DHT_Unified dht(DHTPIN, DHTTYPE);

uint32_t delayMS;

#include <Arduino.h>
#include <Wire.h>

#include <BMP180I2C.h>

#define I2C_ADDRESS 0x77

BMP180I2C bmp180(I2C_ADDRESS);

#include "Ubidots.h"

const char* UBIDOTS_TOKEN = "BBFF-
Trgsxns8oLff9XLmOwuRSgnQbGD97G"; // Put here your Ubidots TOKEN
const char* WIFI_SSID = "voice_wifi"; // Put here your Wi-Fi SSID
const char* WIFI_PASS = "12345678";
const char* DEVICE_LABEL_TO_RETRIEVE_VALUES_FROM =
"40f52016ec29"; // Replace with your device label
const char* VARIABLE_LABEL_TO_RETRIEVE_VALUES_FROM = "control"; //
Put here your Wi-Fi password
Ubidots ubidots(UBIDOTS_TOKEN, UBI_HTTP);

void setup() {
  Serial.begin(115200);
```

```

dht.begin();
Serial.println(F("DHTxx Unified Sensor Example"));

pinMode(D3, OUTPUT);
pinMode(D5, OUTPUT);
pinMode(D6, OUTPUT);
pinMode(D7, OUTPUT);

ubidots.wifiConnect(WIFI_SSID, WIFI_PASS);

sensor_t sensor;
dht.temperature().getSensor(&sensor);
dht.humidity().getSensor(&sensor);
delayMS = sensor.min_delay / 1000;

while (!Serial);
Wire.begin();
if (!bmp180.begin())
{
  Serial.println("begin() failed. check your BMP180 Interface and I2C Address.");
  while (1);
}
bmp180.resetToDefaults();
bmp180.setSamplingMode(BMP180MI::MODE_UHR);

}

void loop() {

  sensors_event_t event;
  dht.temperature().getEvent(&event);
  if (isnan(event.temperature)) {
    Serial.println(F("Error reading temperature!"));
  }
  else {
    Serial.print(F("Temperature: "));
    Serial.print(event.temperature);
    ubidots.add("DHT_Temperature", event.temperature);
    Serial.println(F("°C"));
  }
  // Get humidity event and print its value.

```

```

dht.humidity().getEvent(&event);
if (isnan(event.relative_humidity)) {
  Serial.println(F("Error reading humidity!"));
}
else {
  Serial.print(F("Humidity: "));
  Serial.print(event.relative_humidity);
  ubidots.add("Humidity", event.relative_humidity);
  Serial.println(F("%"));
}

if (!bmp180.measureTemperature())
{
  Serial.println("could not start temperature measurement, is a measurement
already running?");
  return;
}
do
{
  delay(100);
} while (!bmp180.hasValue());

Serial.print("Temperature: ");
Serial.print(bmp180.getTemperature());
ubidots.add("Temperature", bmp180.getTemperature());
Serial.println(" degC");

if (!bmp180.measurePressure())
{
  Serial.println("could not start perssure measurement, is a measurement already
running?");
  return;
}
do
{
  delay(100);
} while (!bmp180.hasValue());

Serial.print("Pressure: ");
Serial.print(bmp180.getPressure());
ubidots.add("Pressure", bmp180.getPressure());

```

```

Serial.println(" Pa");

bool bufferSent = false;
bufferSent = ubidots.send();

if (bufferSent) {
  Serial.println("Values sent by the device");
}
float value1 = ubidots.get(DEVICE_LABEL_TO_RETRIEVE_VALUES_FROM,
  VARIABLE_LABEL_TO_RETRIEVE_VALUES_FROM);

if (value1 != ERROR_VALUE) {
  Serial.print("Value:");
  Serial.println(value1);
}
if (value1 == 1)
{
  digitalWrite(D3, LOW);
  digitalWrite(D5, HIGH);
  digitalWrite(D6, LOW);
  digitalWrite(D7, HIGH);
  Serial.println("Foward");
}
else if (value1 == 2)
{
  digitalWrite(D3, HIGH);
  digitalWrite(D5, LOW);
  digitalWrite(D6, HIGH);
  digitalWrite(D7, LOW);
  Serial.println("Backward");
}
else if (value1 == 3)
{
  digitalWrite(D3, HIGH);
  digitalWrite(D5, LOW);
  digitalWrite(D6, LOW);
  digitalWrite(D7, HIGH);
  Serial.println("Left");
}
else if (value1 == 4)

```

```
{  
digitalWrite(D3, LOW);  
digitalWrite(D5, HIGH);  
digitalWrite(D6, HIGH);  
digitalWrite(D7, LOW);  
Serial.println("Right");  
}  
delay(5000);  
}
```