Chapter 1

Introduction

1.1 Background

A **wheelchair** is used when walking is difficult or impossible due to illness, injury, of disability. Wheelchairs come in a wide variety of formats to meet the specific needs of their users. They may include specialized seating adaption, 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, 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, or by an attendant pushing from the rear.

Though the recent developments of technology and science have drastically changed the way a normal person lives his life. But there are certain groups of people who have disabilities in the lower extreme, for them it is difficult to walk from one place to another. Uneven and rough surface make moving manual wheelchair almost impossible. So we decided to make smart wheelchair, which will make their life easier.

A smart wheelchair aims to provide aid to those handicapped and physically challenged persons by providing them with some sort of mobility which would greatly help them. Smart wheelchair consists of a major controller unit which will allows the user to provide input using a sensor. The controller unit then synthesizes the command and takes required action so as to move the wheelchair in particular position.

There are a wide variety of types of wheelchair, differing by propulsion method, mechanisms of control, and technology used. Some wheelchairs are designed for general everyday use, others for single activities, or to address specific access needs. And new generation of wheelchairs are being developed and used which features the use of artificial intelligence. The project also aims to build a similar wheelchair which would have a sort of intelligence and will help the user on his or her movement.

There are many different types of wheelchairs and we can break them down into four main types. These are manual wheelchair, transport wheelchairs, power wheelchairs and scooters. Then you also get optional bells and whistles in the form of accessories. You also get specialized wheelchairs such as sport wheelchairs, beach wheelchairs, pediatric wheelchairs, bathroom wheelchairs and commercial use wheelchairs.

1. Manual wheelchair

This is one of the type of wheelchair available. Manual wheelchair has 2 large back wheels and 2 front castor wheels and can be self-propelled by the passenger. Manual wheelchairs can be folded and can fit into a normal car. They come fully assembled and the leg rests are removable. It takes lot of energy to move manual wheelchair and plus hands also gets dirty as we have to turn the wheels by the hands.

2. Transport wheelchair:

A transport wheelchair is a wheelchair that is pushed by other person. It has smaller wheels than a standard wheelchairs and it is lighter in total weight. Making it super easy for travelling and lifting into a car. All transport wheelchairs are foldable and compact enough to fit into any car.

Access adaptations such as wheelchair spaces on public transport and wheelchair lifts are frequently designed around a typical manual wheelchair. Power chairs, however, frequently exceed the size and weight limits of manual wheelchairs as they are not constrained by the ability of the user to self-propel. Some designs are too large or heavy for certain wheelchair spaces and lifts. However, there are new designs and innovations seeking to overcome these issues.

1.2 Objective

Wheelchairs are useful for people for whom walking is difficult or impossible due to some illness, injury or disability. There are different types of wheelchairs. Manual wheelchairs are pushed using their handles. Motorised wheelchairs, are driven by joystick.

Voice-controlled wheelchairs are the latest development. These can be driven just by giving voice commands. A more advanced and intelligent version of the wheelchair is controlled directly through human mind.

In case a person is unable to move the wheelchair even with joystick or voice command, an alternative is gesture-controlled wheelchair. The wheelchair moves as per user’s hand gestures. The user has to simply bend or move his/her hand to move the wheelchair.

Some of the Objectives are:

- Wheelchair can be controlled using Hand Gesture.
- The movement of the wheelchair can be controlled using a Sensor.
- Wheelchair will run on motor so manual effort will reduce.
- It will give physically challenged people a better life.
- Our aim is to build a low cost and powerful wheelchair which helps the handicapped people to travel without depending others.
- Introducing a prototype of cost effective electronic gesture based wheelchair
- Easy to operate, because this wheelchair can operated even by a user without palm.

1.3 Need of the Project

The percentage of disabled people has increased in both rural and urban part of India. The disability could be by birth or due to some medical or accidental reason. The aim of this paper is to make a hand gesture controlled wheelchair using accelerometer as sensor to help the physically disabled people in moving from one place to another just by giving direction from the hand. Today in India many people are suffering from disability, there are people whose lower half of the body is paralyzed. This Wheelchair will add on to the comfort and make the life of people bit easier. Around 5436604 people are affected from movement disability. Percentage of population which suffers from different disabilities is shown in graph below. Out of total disability maximum people suffers from disability in movement.

Benefits to people who are:

a. Paralytic person.

b. Those who crawl.

c. Those who walk with the help of aid.

d. Those have acute and permanent problems of joints/muscles.

e. Those who have stiffness or tightness in movement or have loose, involuntary movements or tremors of the body or have fragile bones.

f. Those who have difficulty in motor cell and neurons coordination.

g. Those who have lost sense of sensation in lower part of the body due to paralysis or other problems.

h. Those who have twisted body parts and suffer from any kind of deformity in the body.

1.4 Scope

The smart wheelchair is very useful for physically challenged people. Now a days Development promises a wide scope in developing smart wheelchair. In build voice and Bluetooth function are used to control the wheelchair as well as by using smart phone reading SMS, E-mail, news. In future mind control wheelchair can be developed.

By using intelligent system technology, the smart wheelchair can be created. The joystick control wheelchair is very easy to use. Using joystick people can control the movement of wheelchair very conveniently.

Gesture controlled wheelchair is also very easy to use, as the person just has to move his hand with sensor and then the wheelchair moves in desired directions.

1.5 Organisation of Project

Chapter 1 – Introduction

In this chapter, the basic introduction is given. The background work of the project has been highlighted. It has been concluded in this chapter that, there are many types of wheelchairs and they has some pros and cons. So we have given our idea of providing smart wheelchair which will provide necessary features like Accelerometer sensor to move wheelchair effortlessly.

Chapter 2 – Survey of Technologies

In this chapter, all the necessary technology which we are going to use in our project is given. The hardware and software technologies are used in making this smart wheelchair is given and it is concluded and justified why we have used the technologies to complete the project. In this description and features of software and hardware is given.

Chapter 3 – Requirement analysis

In this chapter, the main problem is defined. And that is divided into subcategories. Then the detailed requirement specification is done. Planning and scheduling of the project is given in this chapter. The hardware requirement and software requirements which will be used to make our smart wheelchair is given. At last conceptual model of our wheelchair is also given which explain the problem domain.

Chapter 4 – System design

In this chapter, desired features and operations are given. The whole problem is divided into manageable parts and each part or module developed separately. And then all the modules are integrated in one. So functionality of all modules is given briefly in this chapter. And also data design, procedural design, data design and logic diagrams also provided.

Chapter 5- Implementation and Testing

In this chapter, implementation approaches and code are given. Define the plan of implementation, and the standard of approach that has been used is explained. How the code is efficient and how we have handled code optimization. The testing approach should be according to the scheme presented in the system design chapter and should follow some approach of testing.

Chapter 6- Result and Discussion

In this chapter, Test Report and User documentation are given. Explain the test result and reports based on the test cases, which should show that the project is capable of facing any problematic situation and that is works fine in different conditions. User documentation defines the working of the software; explain its different functions, components with screen shots. The user documents should provide all the details of the product in such a way that any user reading the manual is able to understand the working and functionality of the document.

Chapter 7- Conclusions

In this chapter, limitation of the system and future scope of project is also given. The conclusion is summarized in fairly short chapters. This chapter brings together many of the points that would have made in the other chapters. Limitation explains the limitations encountered during the testing of the project. Future scope includes new areas of investigation prompted by developments in the projects.

Chapter 2

Survey of Technologies

2.1 Software Technologies

Arduino IDE

The Arduino Integrated Development Environment(IDE) is a cross platform application that is written in the programming language Java. It is used to write and upload programs to Arduino board.

The source code for the IDE is released under General Public License, version 2. The Arduino IDE supports the language 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 stud main() into an executable cyclic executive program with the GNU tool chain, also included with IDE distribution. The arduino IDE employs the program argued 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 boards firmware.

The IDE environment is mainly distributed into three sections

1. Menu Bar

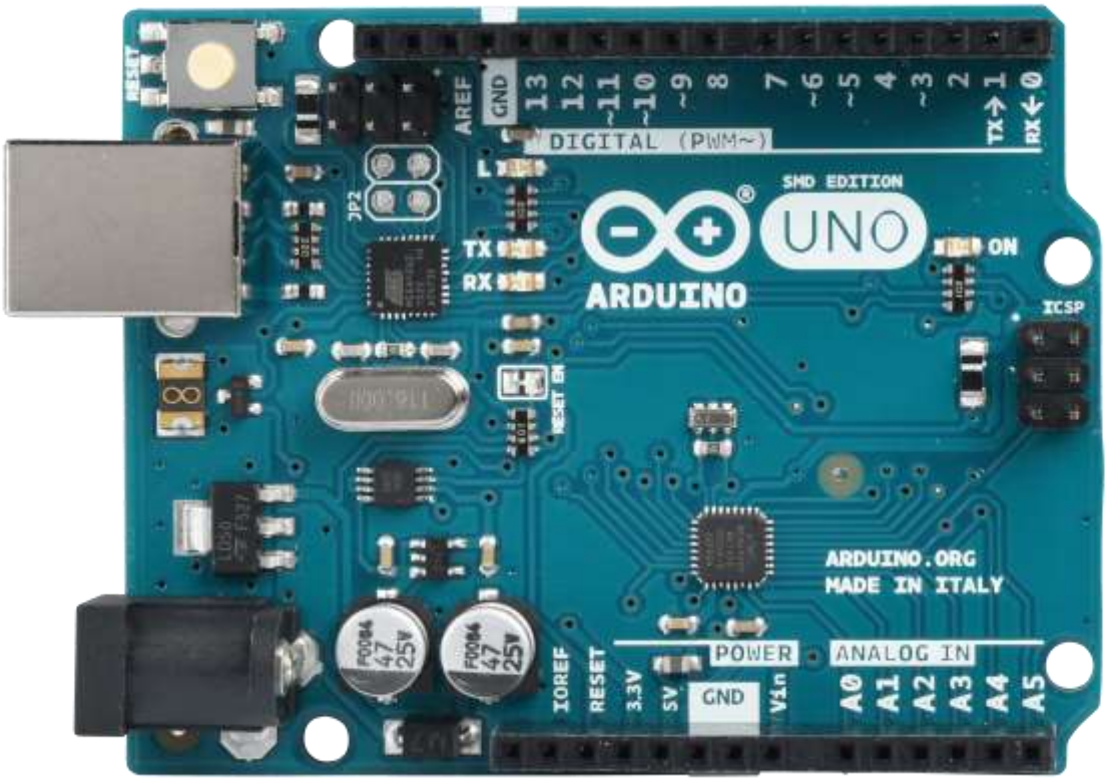
2. Text Editor

3. Output Pane

The digitalRead and digitalWrite commands are used for addressing and making the Arduino pins as an input and output respectively. These commands are text sensitive i.e. you need to write them down the exact way they are given like digitalWrite starting with small “d” and write with capital “W”.

2.2 Hardware Technologies

2.2.1 Arduino UNO



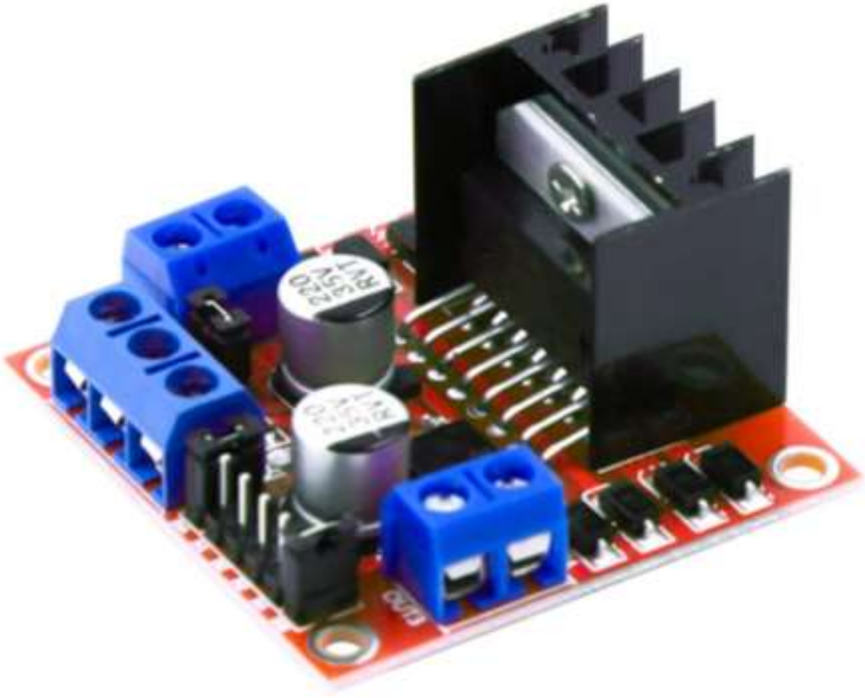
The Arduino Uno board is a microcontroller based on ATmega328. It has 14 digital input/output pins, out of which 6 can be used as outputs, a 16 MHz ceramic resonator, an ICSP header, a USB connection, 6 analog inputs, a power jack and a reset button. This contains all the required support needed for microcontroller. In order to get started, they are simply connected to a computer with a USB cable or with an AC to DC adapter or battery. Arduino UNO board varies from all the other boards and they will not use the FTDI USB-to-serial driver chip in them. It is featured by the ATmega16U2 programmed as a USB-to-serial converter.

Features of Arduino Uno Board:

- It is an easy USB interface. This allows interface with USB as this is like a serial device.

- It is easy to find the microcontroller brain which is a ATmega328 chip. It has more number of hardware features like timers, external and internal interrupts, PWM pins and multiple sleep modes.
- It is an open source design and there is an advantage of being open source is that it has a large community of people using and troubleshooting it. This makes it easy to help in debugging projects.
- It is a 16 MHz clock which is fast enough for most applications and does not speeds up the microcontroller.
- It is very convenient to manage power inside it and it had a features or built-in voltage regulation.
- It has a 32 KB of flash memory for storing your code.
- An on-board LED is attached to digital pin 13 to make fast the debugging of code and to make the debug process easy.
- It has a button to reset the program on the chip.

2.2.2 Motor Driver L298N



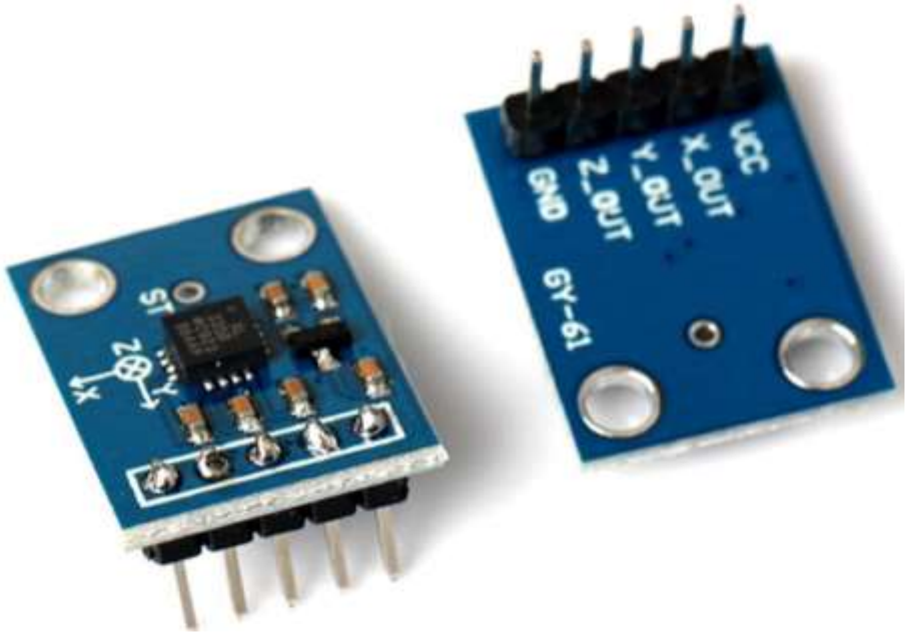
The L298N is an integrated monolithic circuit in a 15- lead Multiwatt and PowerSO20 packages. It is a high voltage , high current dual full-bridge driver de-signed to accept standard TTL logic level sand drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the in-put signals The emitters of the lower transistors of each bridge are connected together rand the corresponding external terminal can be used for the connection of an external sensing resistor. An additional Supply input is provided so that the logic works at a lower voltage.

Features of Motor Driver L298N:

- High operating voltage, which can be up to 40 volts.
- Large output current, the instantaneous peak current can be up to 3A.
- Two built in H-bridge, high voltage, large current, full bridge driver, which can be used to drive DC motors, stepper motors, relay coils and other inductive loads.
- Using standard logic level signal to control.
- Able to drive a two-phase stepper motor or four-phase stepper motor, and two-phase DC motors.
- Adopt a high-capacity filter capacitor and a freewheeling diode that protects devices in the circuit from being damaged by the reverse current of an inductive load, enhancing reliability
- The module can utilize the built-in stabilivolt tube 78M05 to obtain 5v from the power supply. But to protect the chip of the 78M05 from damage, when the drive voltage is greater than 12v, an external 5v logic supply should be used.
- Drive voltage: 5-35V; logic voltage: 5V.
- PCB size: 4.2 x 4.2 cm.

2.2.3 GY61 Accelerometer Sensor

Three sensors are at right angles to one another and measure the acceleration in each of three axes. According to the datasheet for the ADXL-335 they can measure accelleration up to a minimum of +/- 3 g (where 1 g is approximately the equivalent of the acceleration due to gravity on the earth's surface). The sensor outputs an analogue signal for each of the three axes, the voltage is linearly proportional to the acceleration, with 0V being the maximum negative value. The sensor accepts a wide range of supply voltage from 1.8 to 5V, but is designed for 3.3V, so that's what I used.



As each sensor is different they will need some calibration before use. I couldn't figureout a way to reliably calibrate it automatically, but I have a method that you can use tocalculate the range and offsets using a calibration sketch and these values can then be use for more practical applications, or you could have a calibration on start up.

The sensor has a Self Test facility and some breakout boards have a pin marked ST (on the GY-61 there isn't a pin, but there is a hole on the board to the left of the sensor). The ST will exercise the accelerometers to their maximum values. I am not really sure how useful this would be for calibration if you do not know what that maximum actually represents.

The Pins on the GY-61 are:

- Vcc - 3.3V - Red
- Xout - A0 - White
- Yout - A1 - Yellow
- Zout - A2 - Blue
- GND - GND - Black

2.3 Programming Language

- The Arduino integrated development environment (IDE) is a cross-platform application written in java, and it is derived from the IDE for the processing programming language and the wiring projects.
- The Arduino Uno board can be programmed with the Arduino software.
- Select “Arduino Uno from the tools” Board menu.
- The ATmega328 on the Arduino Uno comes returned with a boot loader that allows you to upload new code to it without the use of an external hardware programmer. It communicate using the original STK500 protocol.
- Arduino IDE is a special software running on your system that allows you to write sketches (synonym for program in Arduino language) for different Arduino boards.
- The Arduino programming language is based on a very simple hardware programming language called processing, which is similar to the C language.
- After the sketch is written in the Arduino IDE, it should be uploaded on the Arduino board for execution.
- The first step in programming the Arduino board is downloading and installing the Arduino IDE.
- The structure of Arduino program is pretty simple. Arduino programs have a minimum of 2 blocks,

  Preparation & Execution

- Each block has a set of statements enclosed in curly braces.
- After writing the code we must deploy that code to the Arduino Uno to get the desire output.

Chapter 3

Requirement and Analysis

3.1 Existing System

3.1.1 Description of the Existing system

The present technologies make use of geared motors to move wheelchair in various directions. This is called powered wheelchair which uses battery or power to move wheelchair from one place to another. These systems are controlled using accelerometer sensor as well as hand. These systems are provided with normal braking systems.

There are four main types of wheelchair available in the market. These are Manual wheelchair, Transport wheel chair, power wheel chair, and scooters.

Manual and Transport wheelchair are type of device that a person must move themselves without use of battery. These wheelchairs can be moved by the person or he/she must have a person to push the wheelchair.

Today technology has come up with so much variety and so many features included in wheelchair.

3.1.2 Limitation of existing system

Manual wheelchair:

- Man power is needed to move this wheelchair. So sometimes it becomes difficult to move the chair.
- Person’s hands gets dirty due to pushing wheels by hands.
- Transport wheelchair:
- One always needs a person behind to push chair and have to depend on that person.
- Powered wheelchair:
- Power chair, sometimes, exceed the size and weight.
- It is very expensive. Everyone can’t afford this wheelchair.

3.2 Proposed System

3.2.1 Description of the proposed system.

We propose to develop a smart power wheelchair, whereby the term “smart” means a power wheelchair whose motion is mediated by a computerized system which is aware of the environment and can collaborate with the user to achieve mobility goals and avoid dangerous situation. We plan to develop a wheelchair which is controlled by Accelerometer Sensor.

It is very easy to use. For the connection of Accelerometer Sensor we will use Arduino Uno. We will use arduino for connection and for the coding because it is less costly. And also because arduino is easily available in the market. We could do this by raspberry pi also but availability of raspberry is very less and is little costly than arduino. So arduino is the best option to complete out task. And in we can have better understanding of project. Here we are going to use Accelerometer Sensor as the main input and easy to use this kind of wheelchair.

3.2.2 Advantages of the proposed system

- Cost effective as compared to others.
- Accelerometer Sensor is very easy to use.
- Can be controlled by Hand Gestures.
- Anyone can use this smart chair.
- Light weight.
- Uses less power
- Durability is also very strong.

3.3 Requirement Analysis

3.3.1 Problem definition

Many people are suffering of temporary or permanent disabilities due to illnesses or accidents. For cases of difficult or impossible walking, the use of a wheelchair is becoming essential. Manual or electrical wheelchairs are satisfying for most of the low and medium level disability case where patients can use the wheelchair independently. However, it is difficult or impossible to use wheelchairs independently. In such cases wheelchair users often lack independent mobility and rely on somebody else handle the wheelchair. Researchers involved in wheelchair are aiming at designing smart wheelchairs to solve such problems. This project is to review the recent studies on smart wheelchair systems. It aims to evaluate the current available technologies (Accelerometer Sensor) and to discuss new future directions for our ongoing research.

A handicapped person which locomotive disabilities needs a wheelchair to perform many function to more one place to another. The wheelchair can do so manually by pushing the wheelchair with his hand, hence it is desirable to provide them with a motorized wheelchair that can be moved using Accelerometer Sensor. The motorized wheelchair can move fast and more speed compare to another. The smart wheelchair is achieved at a cost that is affordable for as many handicapped people.

3.3.2 Requirement Specification

Functional requirements

- The wheelchair must provide a comfortable seating in wheelchair
- This system provides many options (Accelerometer Sensor) for handicapped people in one system.
- The wheelchair is also capable to move using Hand Gestures.
- By using this kind of Accelerometer Sensor in wheelchair, saves users a lot of energy.
- In that user very easily handled Accelerometer Sensor for moving in any direction.

- User first priority is that their wheelchair will work fast and at low cost.
- In that main functional requirements in our system are Accelerometer Sensor control work comfortable for handicapped people.

Non-Functional requirements

- Generally informally stated, often contradictory, difficult to enforce during development and evaluate for the customer prior delivery.
- In hardware system that describe the software performance requirement, hardware/software interface requirement, design constraint, and software quality attributes.
- The Accelerometer Sensor system connected to wheelchair in that handicapped people have five option controlling the wheelchair.

1. Moving forward
2. Moving backward
3. Moving left
4. Moving right
5. Stop

- In this system, Accelerometer Sensor will also control the wheelchair using Hand Gestures.

3.4 Hardware Requirements

3.4.1 Motors

- DC motor
- Motor driver L298N
- DC gear motor

3.4.2 Module

- GY61 Accelerometer Sensor

3.4.3 Ardunio

- Arduino UNO

3.4.4 Capacitor & other hardware requirements

- ATMega 328P
- Capacitor
- Registers
- LED lights
- 12V Battery
- Wires
- Switches
- Wheels
- Plastic Cahir

3.5 Software Requirements

- **Ardunio IDE**

Ardunio consists of both a physical programmable circuit board (often referred to as microcontroller) and a piece of software. Ardunio is an open source based on flexible, easy-to-use hardware/ software.

3.6 Justification of Platform

3.6.1 Features of Arduino IDE

1. The Arduino IDE it is Multi-platform Application.
2. Arduino IDE works on the three most popular operating system: Windows, C++, and Linux.
3. The Arduino IDE allows third-party hardware support.
4. With Arduino IDE, users can create programs called sketches that are built with a text editor. The process is a straightforward one though it has several bells and whistles that make the experience more interactive.
5. Arduino IDE has more than 700 libraries integrated. There were written and shared by members of the arduino community that other users can utilize for their own project without having to install anything.
6. In another features are serial monitor, port menu, auto format, project documentation etc.

3.6.2 Advantages of Arduino IDE

1. Very easy to get started.
2. Very easy to extend it and has tons of user contributed shields and libraries. Shields are available to do pretty much anything.
3. Can be used to far real-time application.
4. Everything (both hardware and software, IDE) are open source.
5. Not much programming knowledge needed to do basic stuff.
6. The biggest advantage of Arduino is its ready to use structure.
7. Another big advantage of Arduino is its library of examples present inside the software of Arduino.

Chapter 4

System Design

4.1 Basic Modules

A module is a collection of source files and built in settings that allow us to divide our project into discrete units of functionality. Our project can have one or many modules and one module may use another module. We divide our project into modules to make it easy to understand. And then each module can be independently built, tested and executed.

Basic Module is those that send altered or entirely new request to the crawler and register callbacks to handle the responses. It is a self-contained component in hardware project. When the connections of both the modules are done then we can integrate them in one.

Module: GY61 Accelerometer Sensor

Sensor is turned to exact middle position. Till the Sensor is kept at middle the motor will be stop. When the Sensor is moved the potentiometer encodes analog voltage values and transfers it to the Arduino board through the analog data pin.

The Sensor is an analog device that uses 2 potentiometer to give analog values, one for each direction. The two directions is just the X and Y direction, and one potentiometer is meant to be for one direction. Hence for one position of sensor shaft, one for x and another for y. So for any position of sensor we have 2 values, and we will write code to read those values and used different conditions (like up, down, left, right) to send HEX values.

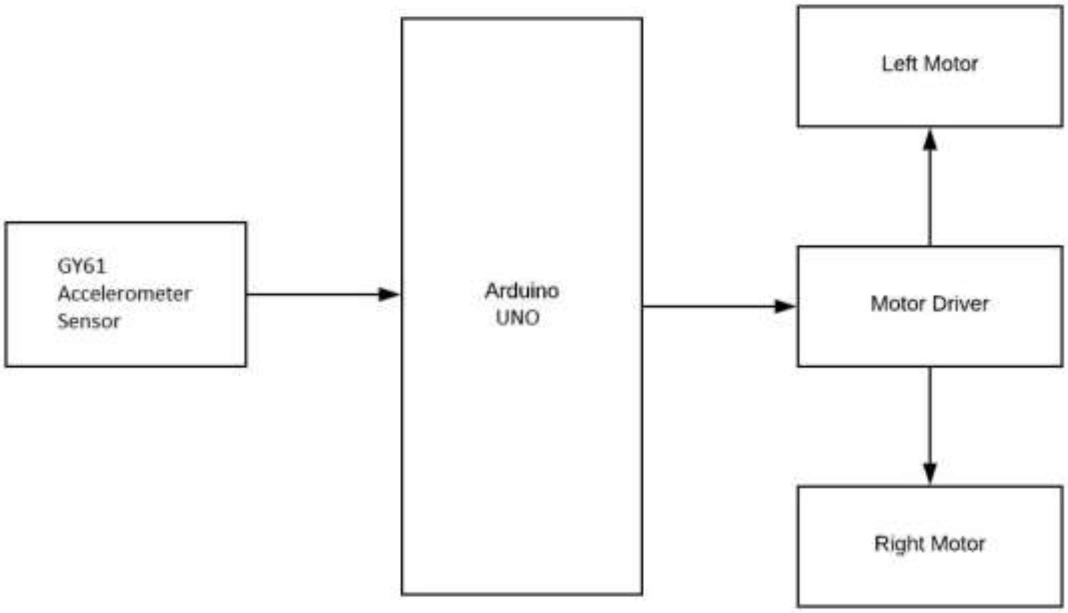


Fig. Accelerometer Sensor Module

4.2 Procedural Design

Procedural design is nothing but the control commands on the predefined data or information. Project has their own system and control command on that functional model of the project which is done by procedural design. In the procedural design it includes the set of specified instruction of the directions.

There data entity like it have repetition, sequence, and the conditional predefined information for the system. It is a collection of the sequence algorithms which is also called as Frame work of the project analysis. It serves the action of the specified algorithms which is based on the processing steps of the project so that the process for the procedural design structure will be in the proper phase.

The project report is handed by the procedural design with the methodology evaluation. It includes the initial phases of each of the process which is use in the algorithm.

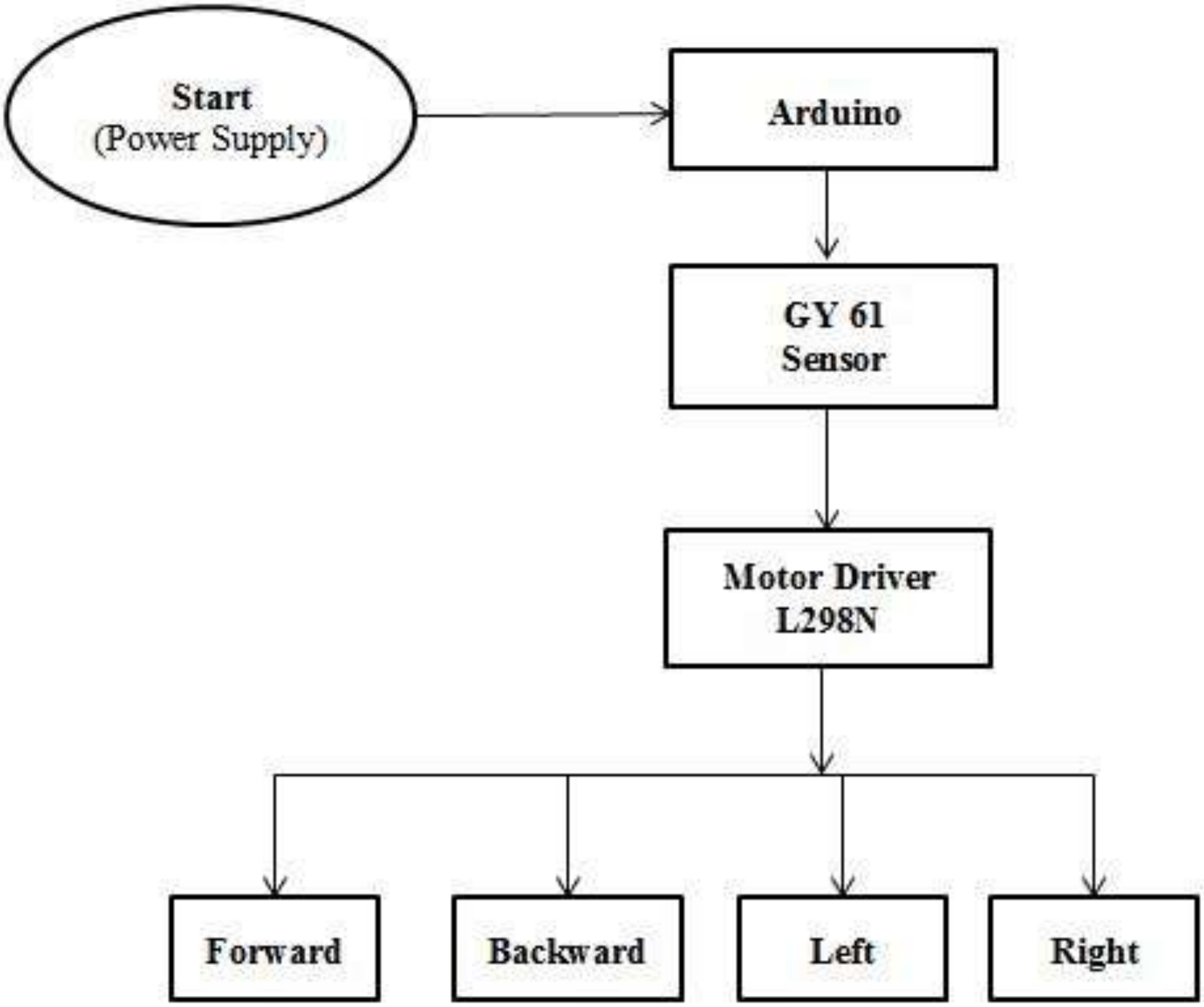


Fig. Procedure diagram

4.2.1 Logical Diagram

In logical basic diagram for the project structure using that we can determine the project data flow. Basically it is used of the small scale purpose requirement specification. Here we are use logical diagram for the project that is how the process should be performing in their requirement analysis.

Logical diagram is used to perform what action should be taken by the people. It is describes that what events should take place and the information required for each event to perform the process. When there is long scale level project at that time logical diagram will help to differentiate with different types of form of the project system. The logical diagram use to implemented project logic, pass the logical command.

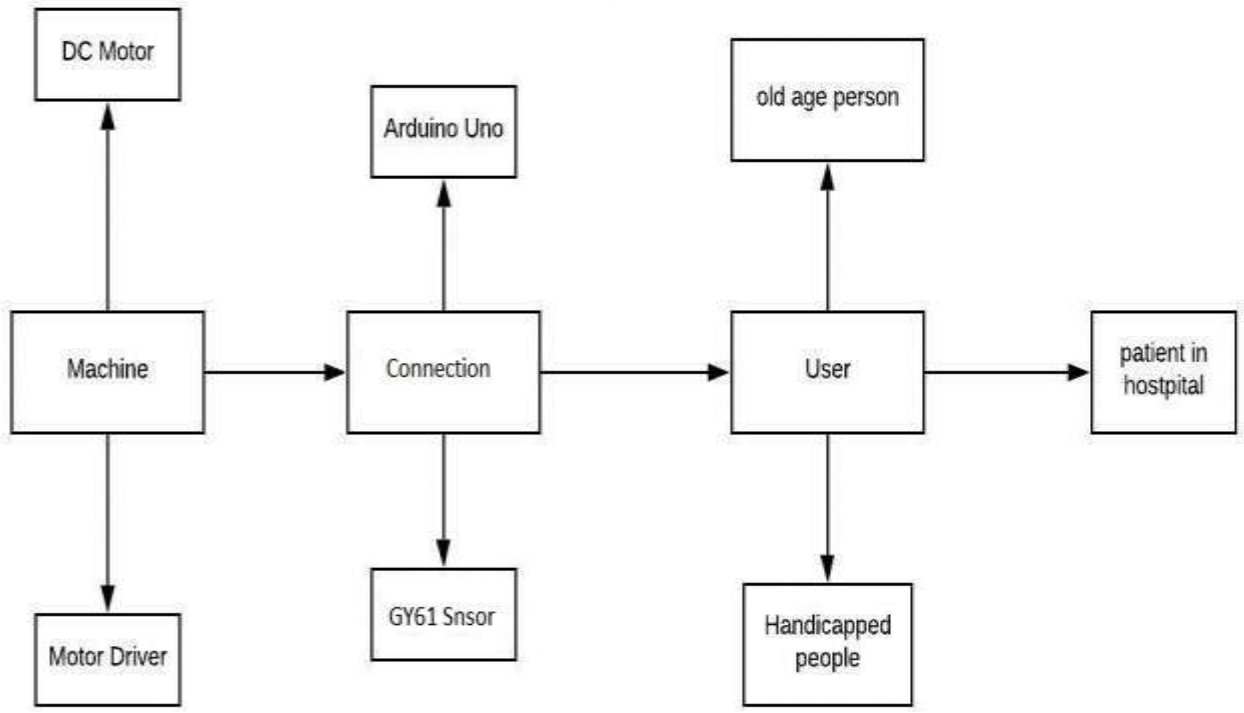


Fig. Logical diagram

4.2.2 Flow Chart

Flow chart is used to design the flow of the project, process and design the documentation of the system. It is used to design the simple program on the small level of the system. Flow chart is used to define the process of action which is going perform on the given input. We can simply say that it is one type of the chart where the instructions will be performed of the process where the instructions will be performed.

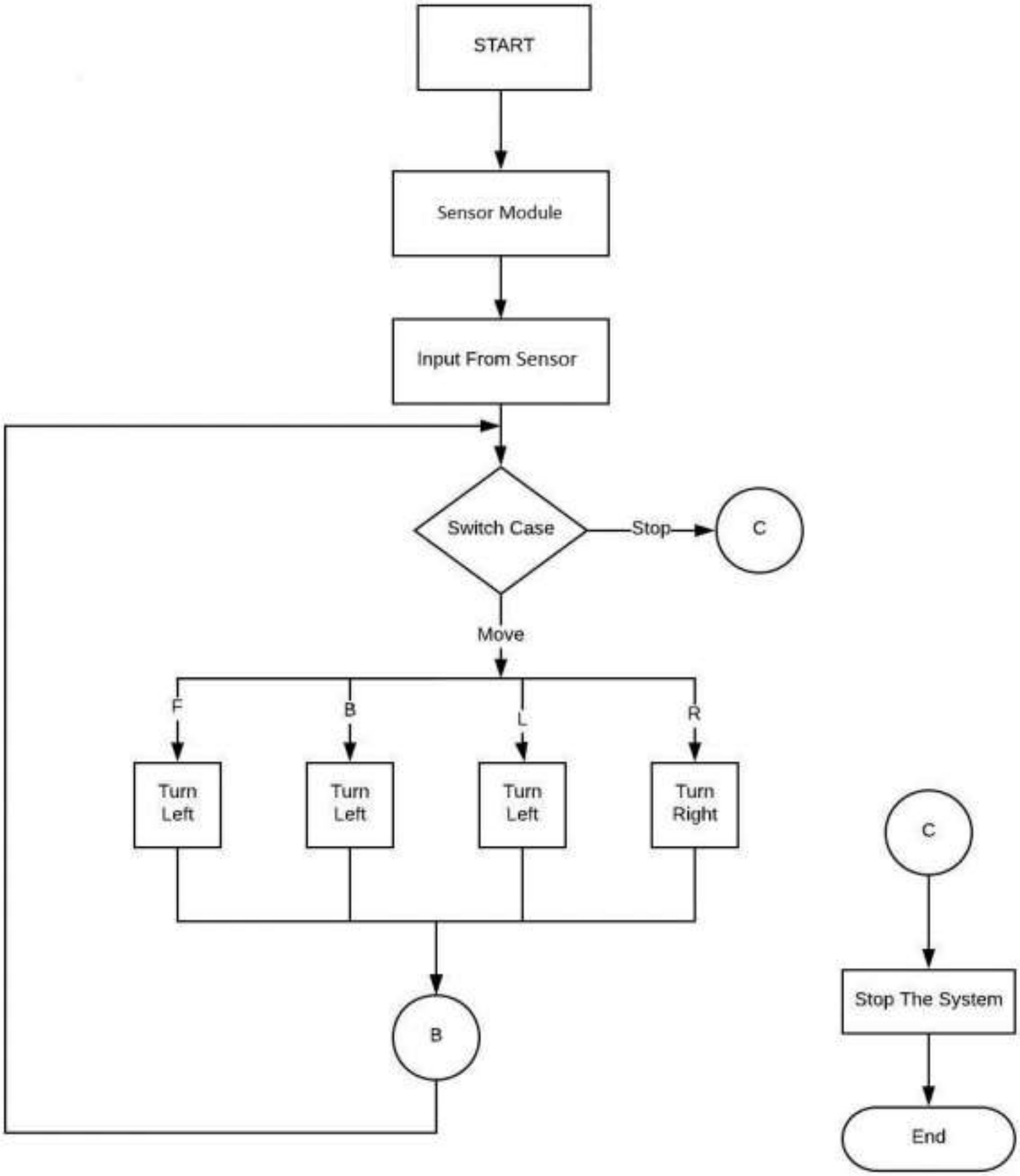


Fig. Flow chart

4.3 Data Structure Diagram

The data structure diagram important part of the system because they represented the how flow the data in system. For keeping the data and place it on the efficient location data structure diagram is used.

As we know that the data is a collection of information, data values, relationship between data organization, it is also used to functional operations which is applied on the data structure.

It is used to define the data on every action of the process. Where the data should be place and how the data will be access by the client (handicapped people) all this tasks is handle by the data structure diagram.

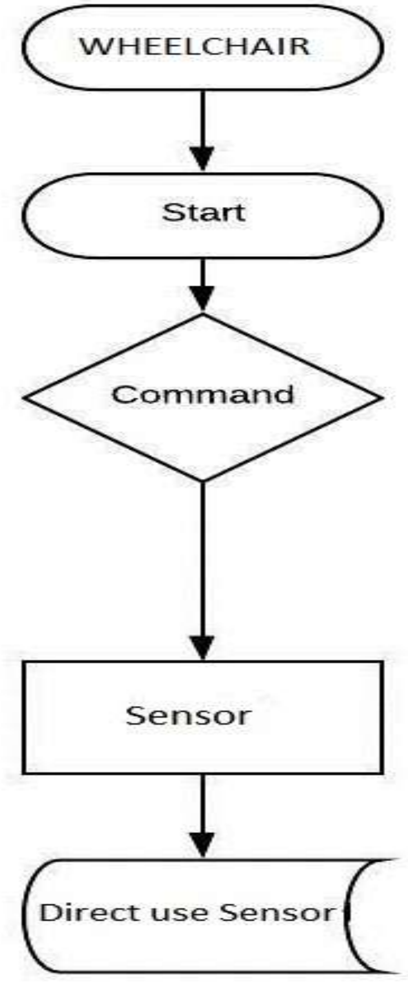


Fig. Data structure diagram

4.4 Algorithm Diagram

Algorithm Diagram is a set of action to be perform for the determining or we can also say that solving the problem of our system. It is a predetermining instruction which is set for the particular problem for their execution. When we use algorithm diagram there is a set of steps which we have to follow that. In the given diagram it is a set of instruction which is done by the flow chart and algorithm diagram.

In the algorithm Diagram first we define the steps which we have to perform our system.

Step 1: Start the process

Step 2: Initialize the system components

Step 3: Take the command

Step 4: Check the given command

Step 5: If true then execute the next step otherwise jump on again Step 3

Step 6: End the program execution

It is a set of command taken by the users and execute the program using algorithm. Program run using instruction set. Basically we use algorithm because of run the every command/instruction step by step.

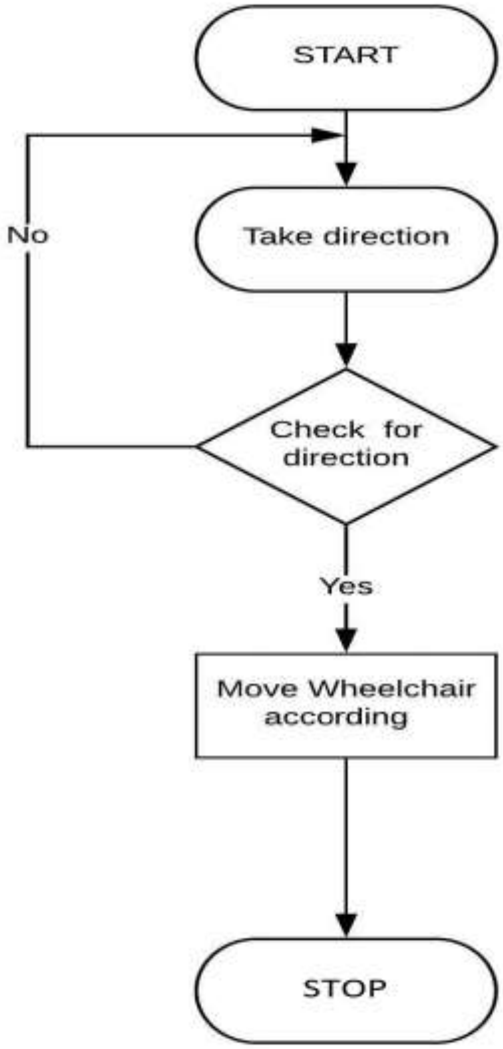


Fig. Algorithm Diagram

4.5 Test Case

A test case is a set of action executed to verify to your particular feature or functionality of your hardware project. every project there is need of Test Case because it helps to give the proper specification of the given input on the basis of the input what condition should be perform, how the execution will take places in the process execution. When command will give to the input taker then what action to be perform and according to that what is the expected result will come from the action taken. It is a technique case for verifying all the aspect of the system.

Chapter 5

Implementation and Testing

5.1 Implementation Approaches

5.1.1 Software Implementation

In software development, the **V-model** represents a development process that may be considered an extension of the water fall model, and is an example of the more general V-model. Instead of moving down in a linear way, the process steps are bent upwards after the coding phase, to form the typical V shape.

The V-Model demonstrates the relationships between each phase of the development life cycle and its associated phase of testing.

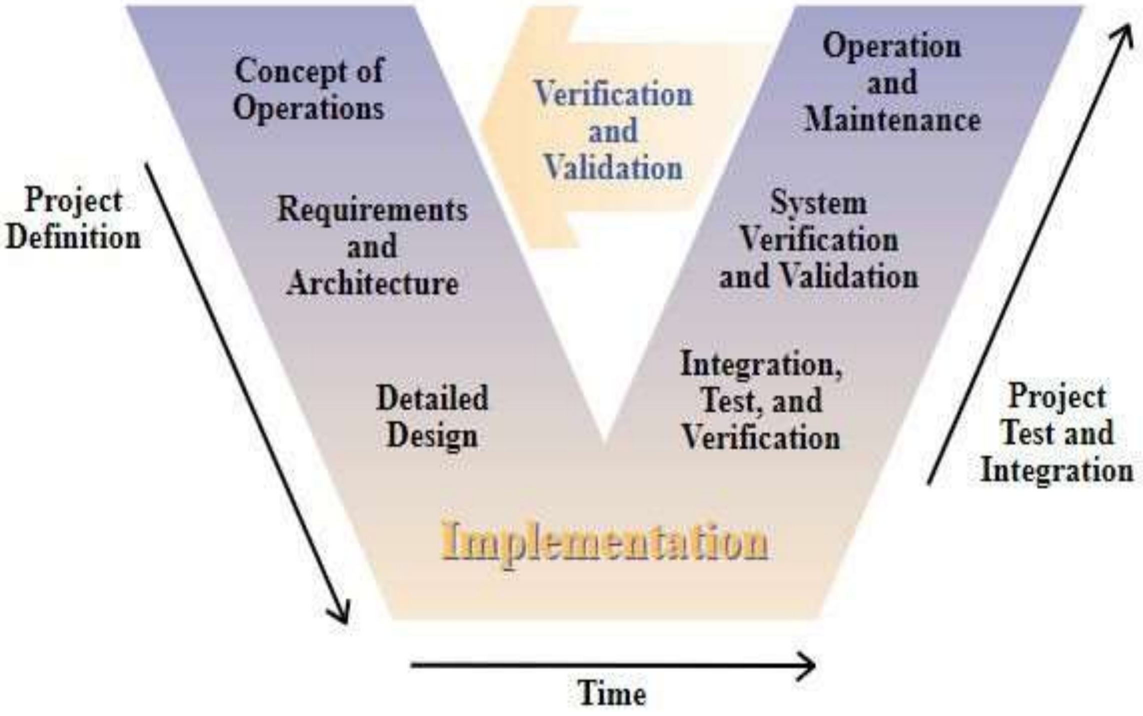

Concept of Operations
Verification and Validation
Operation and Maintenance
Project Definition
Requirements and Architecture
System Verification and Validation
Detailed Design
Integration, Test, and Verification
Project Test and Integration
Implementation
Time

Fig. V model

5.1.2 Hardware Implementation

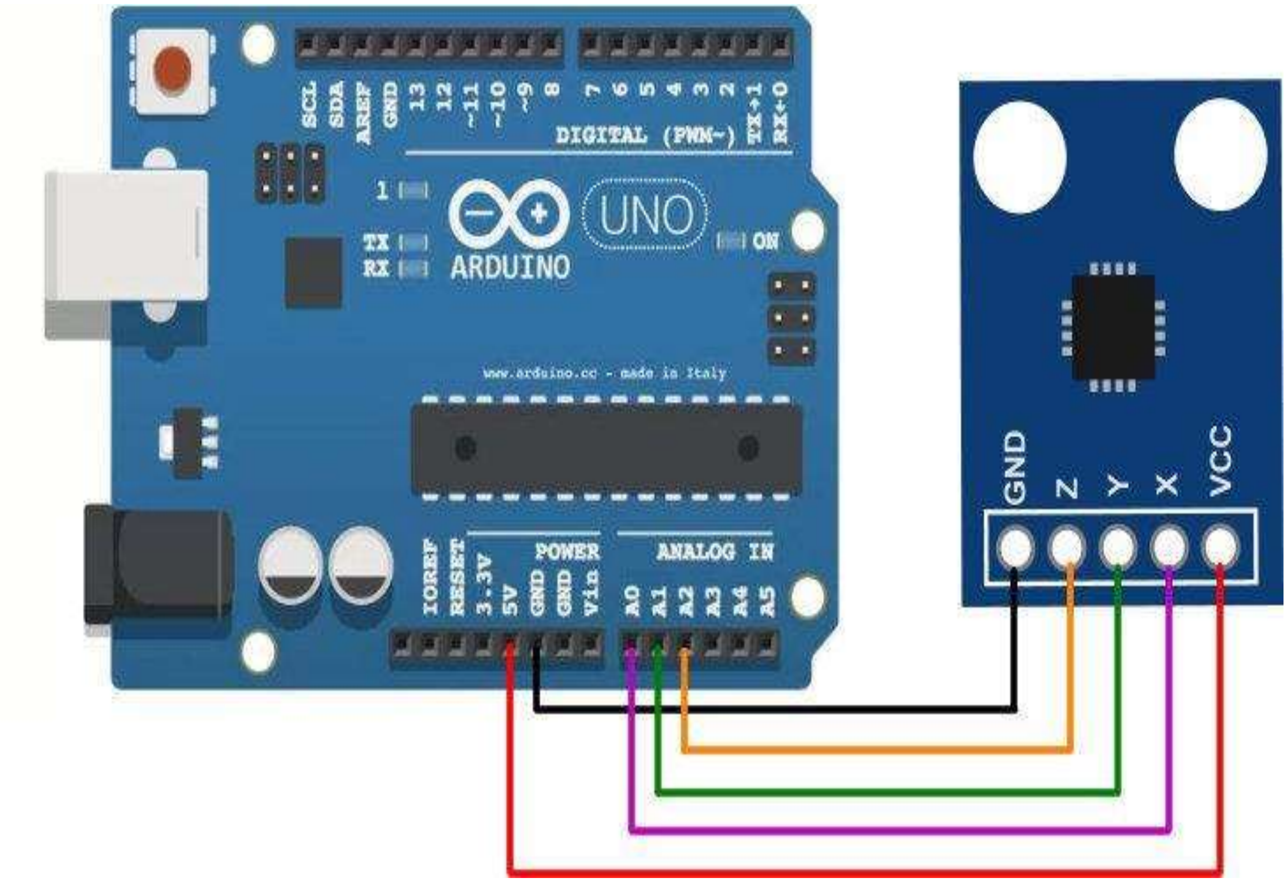


Fig. GY61 Accelerometer Sensor Diagram

In this project we have used a Pastic chair for demonstration. This Wheelchair has GY61 Accelerometer Sensor to move the Wheelchair using Hand Gestures. The Accelerometer Sensor system connected to wheelchair in that handicapped people have five option controlling the wheelchair Moving forward, Moving backward, Moving left, Moving right and Stop. If the User Moves the hand up the wheelchair will move Forward. If the User Moves the hand right side the wheelchair will moves Right and so on.

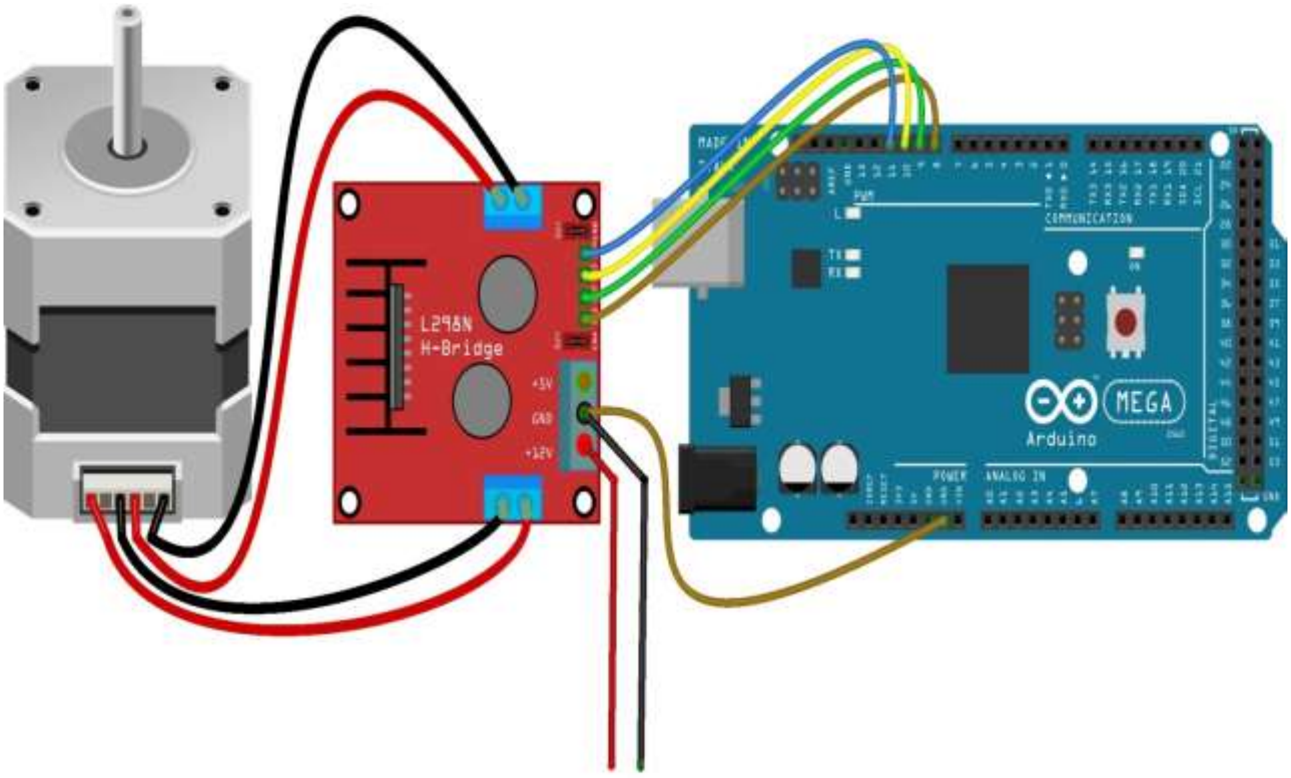


Fig. Motor Driver L298N Diagram

In this project we have used a Pastic chair for demonstration. This wheelchair have two dc motors at its front and back side. Front side motor is used for giving direction to, means turning left or right side. And Back side motor is used for driving the car in forward and backward direction. An Accelerometer Sensor is used to receive command from User and Arduino UNO is used for controlling the whole system.

5.2 Coding

Arudino Code

```
const int xPin   = A0;
const int yPin   = A1;
const int zPin   = A2;
const int in_1 = 8 ;
const int in_2 = 9 ;
const int in_3 = 10 ;
const int in_4 = 11;
const int in_5 = 7 ;
const int in_6 = 6 ;

int X = 0;
int Y = 0;
int x = 0;
int y = 0;

void setup()
{
  Serial.begin(9600);
  pinMode(in_1,OUTPUT) ;
  pinMode(in_2,OUTPUT) ;
  pinMode(in_3,OUTPUT) ;
  pinMode(in_4,OUTPUT) ;
  pinMode(in_5,OUTPUT) ;
  pinMode(in_6,OUTPUT) ;
}

void loop()
{
x = analogRead(xPin);
y = analogRead(yPin);

Serial.print("x = ");
Serial.print(x);
Serial.print("  y = ");
Serial.print(y);
Serial.print("  AngleX = ");
```

```
X = constrain(map(x,380,270,0,180),0,180);
Serial.print(X);
Serial.print("  AngleY = ");
Y = constrain(map(y,380,270,0,180),0,180);
Serial.println(Y);

  if(X>160)
  {
    digitalWrite(in_1,HIGH) ;
    digitalWrite(in_2,HIGH) ;
    digitalWrite(in_3,LOW) ;
    digitalWrite(in_4,HIGH) ;
    Serial.print("   Left   ");
  }
  else if(X<20)
  {
    digitalWrite(in_1,LOW) ;
    digitalWrite(in_2,HIGH) ;
    digitalWrite(in_3,HIGH) ;
    digitalWrite(in_4,HIGH) ;
    Serial.print("   Right   ");
  }
  else if(Y<20)
  {
    digitalWrite(in_1,LOW) ;
    digitalWrite(in_2,HIGH) ;
    digitalWrite(in_3,LOW) ;
    digitalWrite(in_4,HIGH) ;
    //digitalWrite(in_5,HIGH) ;
    //digitalWrite(in_6,LOW) ;
    Serial.print("   Front   ");
  }
  else if(Y>160)
  {
    digitalWrite(in_1,HIGH) ;
    digitalWrite(in_2,LOW) ;
    digitalWrite(in_3,HIGH) ;
    digitalWrite(in_4,LOW) ;
    //digitalWrite(in_5,LOW) ;
    //digitalWrite(in_6,HIGH) ;
    Serial.print("   Back   ");
  }
  else
  {
    digitalWrite(in_1,HIGH) ;
    digitalWrite(in_2,HIGH) ;
    digitalWrite(in_3,HIGH) ;
    digitalWrite(in_4,HIGH) ;
    digitalWrite(in_5,HIGH) ;
    digitalWrite(in_6,HIGH) ;
    Serial.print("   Halut   ");
   }
}
```

5.3 Testing Approaches

A test approach is the test strategy implementation of a project, defines how testing would be carried out. Test approach has two techniques:

- **Proactive -** An approach in which the test design process is initiated as early as possible in order to find and fix the defects before the build is created.
- **Reactive -** An approach in which the testing is not started until after design and coding are completed.

5.3.1 Unit Testing

Unit testing is a level of software testing where individual units/components of software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest test able 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 Benefits

- Unit testing increases confidence in changing/maintaining code. If good unit test are written and if they are run every time any code is changed, we will be able to promptly catch any defects introduced due to the change. Also, if codes are already made less inter dependent to make unit testing possible, the un intended impact of changes to any code is less.
- Codes are more reusable. In order to make unit testing possible, codes need to be modular. This means that codes are easier to reuse.
- Debugging is easy. When a test fails, only the latest change need to be debugged. With testing at higher levels, changes made over the span of several days/weeks/months need to be scanned.
- The cost of fixing a defect detected during unit testing is lesser in comparison to that of defects detected at higher levels. Compare the cost(time, effort, destruction, humiliation)of a defect detected during acceptance testing or when the software is live.

5.4.2 Integrated Testing

Integration Strategies

- **Big Bang Approach:** Big Bang Integration Testing is an integration testing strategy where in all units are linked at once, resulting in a complete system. When this type of testing strategy is adopted, it is difficult to isolate any errors found, because attention is not paid to verifying the interfaces a cross individual units.

- **Top down Integration:** Top-down integration testing is an integration testing technique used in order to simulate the behavior of the lower-level modules that are not yet integrated. Stubs are the modules that act as temporary replacement for a called module and give the same output as that of the actual product. There placement for the' called' modules Is known as' Stubs' and is also used when the software needs to interact with an external system.

- **Bottom up Integration:** Bottom-up testing is an approach to integrated testing where the lowest level components are tested first, then used to facilitate the testing of higher level components. The process is repeated until the component at the top of the hierarchy is tested. All the bottom or low-level modules, procedures or functions are integrated and then tested. After the integration testing of lower level integrated modules, the next level of modules will be formed and can be used for integration testing .This approach is helpful only when all or most of the modules of the same development level are ready. This method also helps to determine the levels of software developed and makes it easier to report testing progress in the form of a percentage.

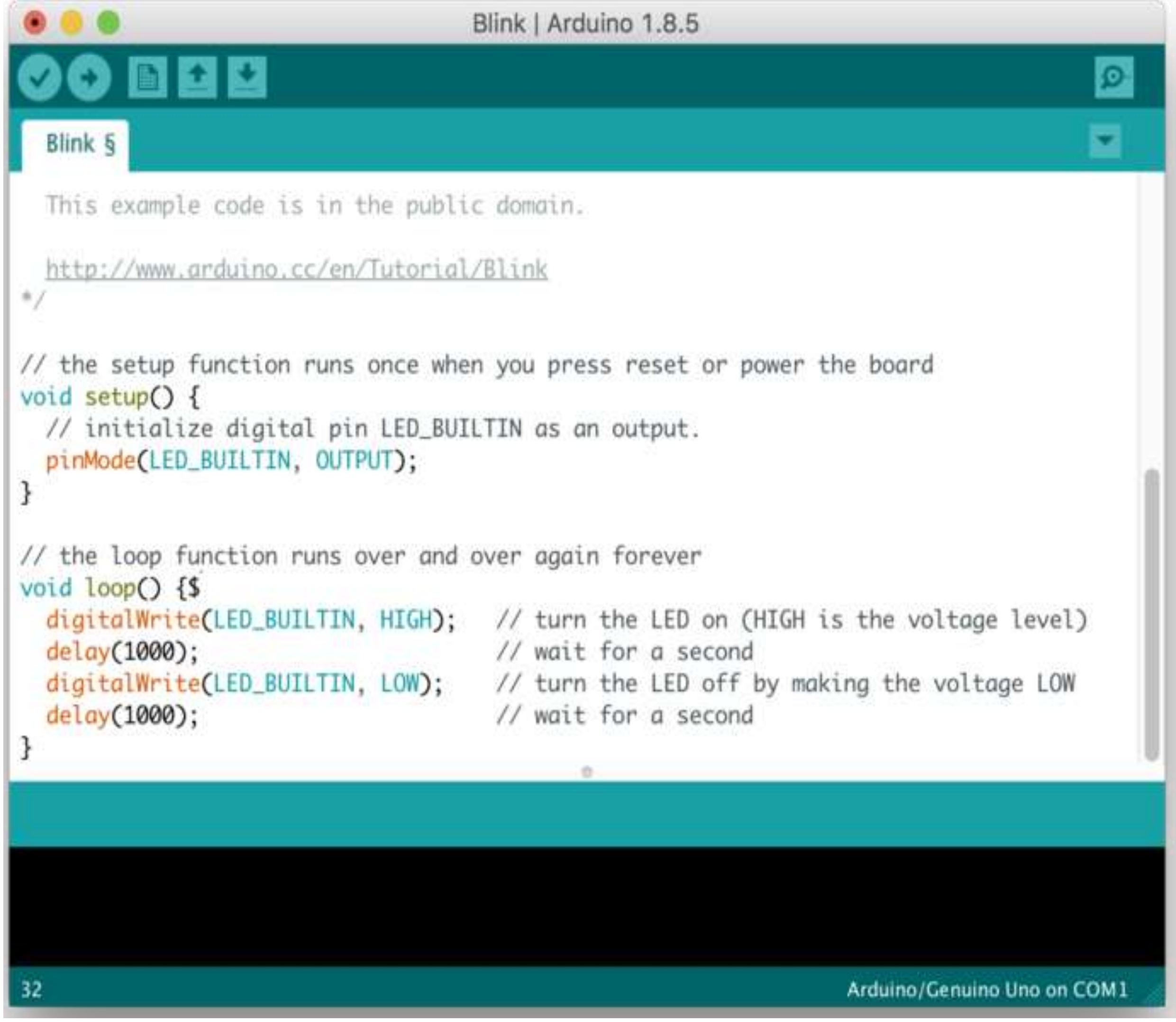
- **Hybrid Integration:** A hybrid integration platform is a combination of on-premise and cloud based system, securely connected using a technology like Transport Layer Security, so that it supports the integration of on-premise end points, cloud end points and the combination of the two, for all integration patterns.

Chapter 6

Result and Discussion

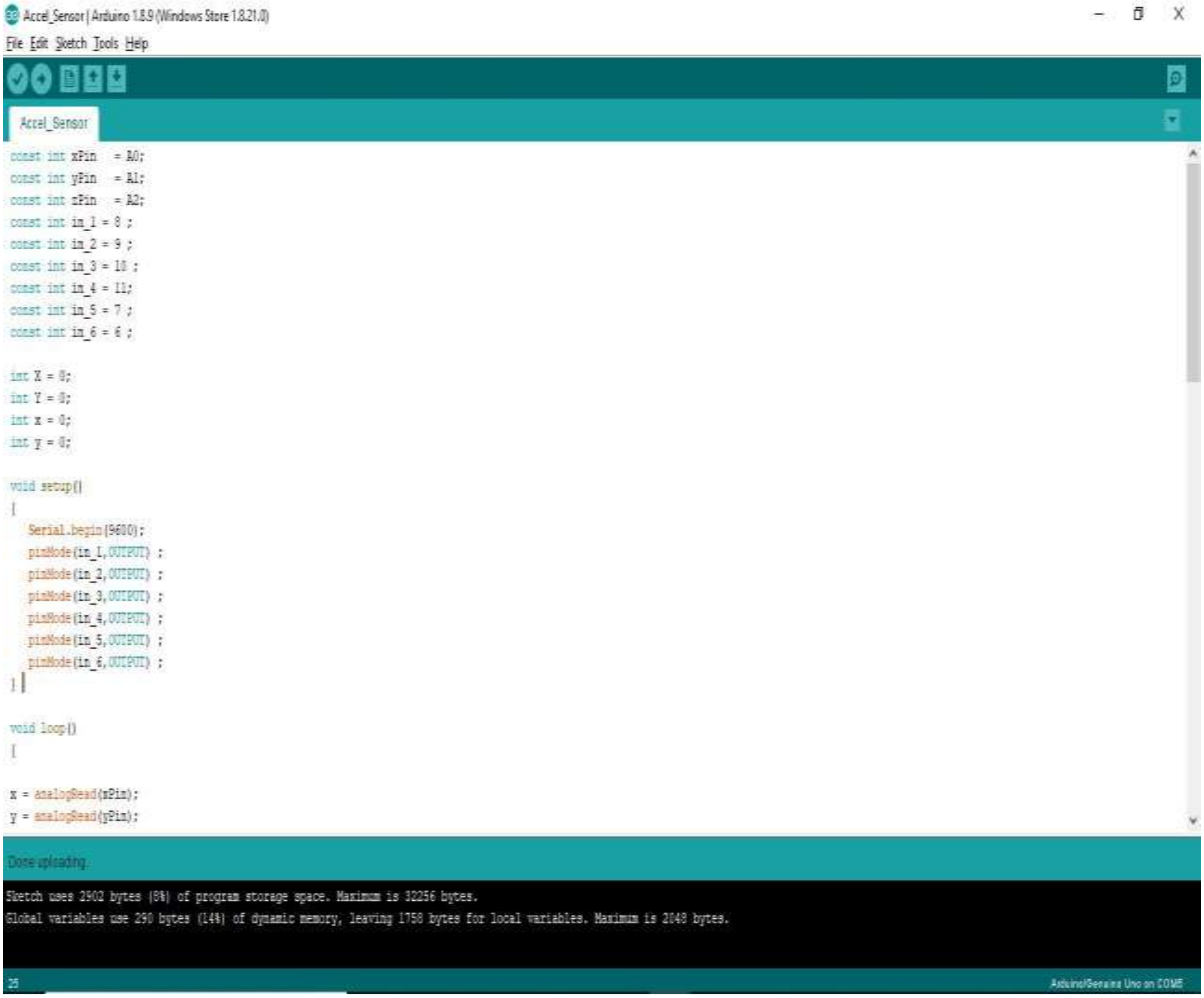
6.1 Test Reports

Frist Screen of Arduino



```
  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Blink
*/

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {$
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);                       // wait for a second
}
```

Code Deployment in Arduino

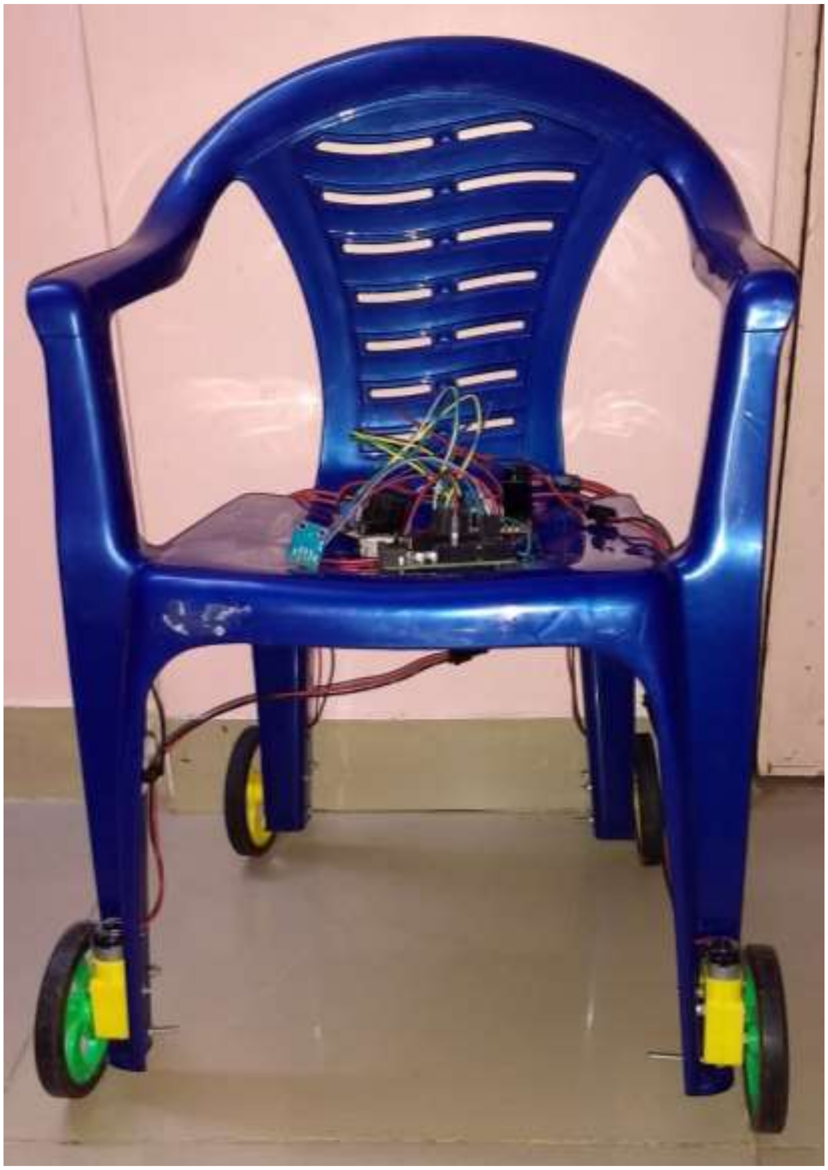

Accel_Sensor | Arduino 1.8.9 (Windows Store 1.8.21.0)
File Edit Sketch Tools Help
Accel_Sensor

```
const int xPin   = A0;
const int yPin   = A1;
const int zPin   = A2;
const int in_1 = 8 ;
const int in_2 = 9 ;
const int in_3 = 10 ;
const int in_4 = 11;
const int in_5 = 7 ;
const int in_6 = 6 ;

int X = 0;
int Y = 0;
int x = 0;
int y = 0;

void setup()
{
   Serial.begin(9600);
   pinMode(in_1,OUTPUT) ;
   pinMode(in_2,OUTPUT) ;
   pinMode(in_3,OUTPUT) ;
   pinMode(in_4,OUTPUT) ;
   pinMode(in_5,OUTPUT) ;
   pinMode(in_6,OUTPUT) ;
}

void loop()
{

x = analogRead(xPin);
y = analogRead(yPin);
```

Done uploading.

```
Sketch uses 2902 bytes (8%) of program storage space. Maximum is 32256 bytes.
Global variables use 290 bytes (14%) of dynamic memory, leaving 1758 bytes for local variables. Maximum is 2048 bytes.
```

6.2 User Documents



Chapter 7

Conclusion

7.1 Conclusion

This project elaborates the design and construction of smart Wheelchair with the help of Accelerometer Sensor. The circuit works properly to move as the command given by user. After designing the circuit that enables physically disabled to control their wheel using a hand gestures and it has also been tested and validated. The detection of any movement is successfully controlled by the microcontroller. This proposed system contributes to the self-dependency of differently able and older people.

In the system design we conclude that all the design of the process model and diagram of the system models are specifically for the process purpose and it is also defining that how the flow of the process takes place. There are many system diagram which is going to help in the process designing and after that how we can implement the system in actual way of the action. Basically all the system diagram and design will have only one purpose how to execute the process and how the flow of the system execution will take place in the given required components and aspect of the project. And we conclude how to command taken by the users and perform by the set of instruction. We have use two type of instruction, this instruction are very easily or comfortable use to users. Finally conclusion is the system is develop are very easy to operate and given instruction easily as well as.

We have designed this wheelchair for the physically disabled people those who cannot walk, so that they can easily handle it with their hands by using the Joystick. But for those people who cannot move their legs as well as hands, the voice recognition control wheelchair can resolve this issue. We can also add a sensor unit to the circuit so that it can detect the obstacle in its path.

7.2 Future Scope

1) **Mind Control**

   Controlling wheelchair by electric signal coming from brain. As our brain contains thousands of neuron.

2) **GPS navigation**

   Navigating the actual position of wheelchair.

3) **Alternate power source**

   Solar panel roof can be used as alternative power source and also can be a protective layer from rain and sun.

4) **Step climbing and Intercommunication between devices**

   It is just upcoming project trying to be implemented on road vehicles. The communication aid with these wheelchairs will help the deaf and dumb to communicate with each other's too.

5) **Bluetooth Control**

   Controlling wheelchair by using Smart phone using Bluetooth.

References

1. https://microcontrollerslab.com/ultrasonic-sensor-working-applications-advantages/
2. https://www.tutorialspoint.com/arduino/arduino_ultrasonic_sensor.htm
3. https://www.researchgate.net/publication/266373564_Smart_Wheel_Chair#pf1c
4. http://softwaretestingfundamentals.com/integration-testing/
5. https://www.tutorialspoint.com/software_testing_dictionary/test_approach.htm
6. http://softwaretestingfundamentals.com/unit-testing/
7. https://create.arduino.cc/projecthub/ammaratef45/detecting-obstacles-and-warning-arduino-and-ultrasonic-13e5ea
8. https://circuitdigest.com/microcontroller-projects/arduino-joystick-interfacing
9. https://circuitdigest.com/microcontroller-projects/bluetooth-controlled-robot-car-using-arduino
10. https://circuitdigest.com/