Academic Year: 2023/2024

Enrolment Number: 30052265

Name: Hamza Afzal

Course: BEng Electrical and Electronics Engineering (Hons)

Scheme: Automatic Guidance System for Safety of the Visually Impaired.

Supervisor : Ben Mehenni

University of South Wales
Prifysgol De Cymru

Abstract – Visually impaired people are those, whose eyes can't see objects due to visual acuity loss or they have a reduced visual field of view[1]. This makes the movement of a person very difficult and restricted. The solution to this problem is a stick to help navigate by avoiding obstacles. But overtime the world has progressed in the technological domain hence there exist many ways now to aid the blind people and overcome this problem, partially and completely as well. One of the most appropriate solutions to this problem is a Smart Stick. This is an improvement made on a traditional method of navigation for blind people. The stick is now equipped with modern technology. This includes a system which is based on Arduino Nano with an ATmega328 microprocessor. The Arduino is coupled with Ultrasonic Sensor to identify the obstacles. The stick also comes with an SOS button, in which the user can press the button in an emergency and the location is sent to the pre-defined mobile number. This is done using a GPS module and a GSM Module.

Keywords: Obstacle Detection, Ultrasonic Sensor, GSM Module, GPS Module, Visual Impairment

[1] (The University of Pittsburgh, 2024)

Contents

Chapter 1
Introduction

Chapter-1: Introduction

One of the senses which humans possess is called Vision. It helps us see and identify objects with the help of our brain. But Vision could be lost due to various reasons and diseases e.g. Cataract. In the UK, 2 million people have sight loss. Almost 340,000 of these are blind or partially sighted[2]. Visually impaired is a broad term as in a sense that there are two categories in it, Low Vision, and Blindness. WHO defines Low Vision as visual acuity between 20/70 and 20/400 whereas Blindness worse than 20/400[3]. Generally, if someone is visually impaired, they might still possess some vision. But it totally depends on how they use the vision they have. This is a factor which is natural, and which varies human to human. But here we will talk about the visually impaired people who are unable to see obstacles or have difficulty in sighting obstacles. This difficulty in a human's life can have many adverse effects on physical, mental and financial health of a person. E.G it is difficult for blind people to walk to a park and exercise which effects both physical and mental health. Similarly, a person will have difficulty going to work.

Hence, the aim of our project is to go over this problem and cater solutions for this. One of the most efficient and effective solutions is to upgrade the tools visually impaired are already using, which means upgrading the stick. The main goal is to make the stick detect an obstacle.

1.1 Aims

The main goal of this project is to provide visually impaired people an economically viable tool to make their daily life easier which can only be done by avoiding obstacles effectively. By economically feasible, we mean that it should be cheaper than the other products which are available in the market but also it means that this should be affordable for all the visually impaired people. But the key here is to not use components which are unreliable to make the project cheaper, but to choose components which are economical and reliable in effectiveness along with being efficient which makes the system work constantly without any hiccups. So, an economical project will be our number one goal.

In Addition of this, we will also aim to make sure the ease of the user. Ease of the user generally is the easiness the user feels whilst using the product. This is really important. The reason behind this generally is if the product is heavy, or difficult to handle or maybe the handle is made of an uncomfortable material, this will result in negative reviews from the user hence we have to make sure that we have to make the user experience as better as possible. This also includes the position of the components, how the main box will be designed, where it will be placed, what it would be made of etc.

The reason we put economy as the number aim was because everything is related to the budget. But this will push us to our limits to explore the right choice of components and materials.

1.2 Initial Planning

The initial plan is to have a stick which can detect obstacles along with it, it should have an emergency or SOS button. When the button is pressed it should send an SMS which will contain the current location of the person. The SMS will be sent to a designated phone number. All the components will be soldered to a PCB which then will be put in a 3D Printed Cover. The cover will be

---

[2] (NHS UK, 2021)
[3] (The University of Pittsburgh, 2024)

mounted on a generic blind stick. The blind stick will also have the sensors attached to it which are used to detect the obstacles.

1.3 Objectives

1. Reliability and Safety
2. Economic Feasibility
3. Obstacle Detection
4. Efficient Communication
5. Location Tracking
6. Power Management

These are some of our main objectives. Reliability and Safety is very important in this project as the end user is visually impaired hence, we have to make sure that the system is reliable enough that it doesn't fail. And if it does it has the capacity and capability to recover from the malfunction.

The project should be able to accurately and consistently detect obstacles at a certain distance. So that the user can be alerted beforehand therefore the user is able to change the direction to avoid the obstacle.

Efficient Communication and Real Time location tracking play a pivotal role in this project. Hence, we have to assure that they are accurate, and they work well in different conditions. We also need to consider the appropriate data rates/protocols, power consumption and signal stability to maintain a stable system.

Power Management is fundamentally important for this project. This is because we wouldn't want the product to turn off in the middle of the pavement. Hence, we need to optimize the design of the electronics. Along with this we need to make sure that the system consumes as much less energy as possible so that we can make the most out of a smaller battery pack. We can also alter the software to make the system efficient when it is idle by having minimum delays and versatility.

Last but not the least, we should make this project in accordance compliance and regulation. This also includes following certain standards which could be local and international. This will increase the trust of the customer on the product and could be one of Unique Selling Point's of the product. These certifications will also lead us to entering different market of the world very easily hence from a business perspective, this will be beneficial.

1.4 List of Equipment/Tools

1. Arduino Nano
2. SIM800L
3. NEO-6M
4. Tactile Button
5. LED's
6. Ultrasonic Sensors
7. Lithium Polymer Batteries
8. TP4564 Charging Module
9. Arduino IDE
10. Internet

1.5 Hardware

Arduino Nano is a compact development board. It comes with a micro controller on board. It is developed by Arduino LLC. It is very popular among people who work in IOT field because of its compact form factor. This is also preferred because it has built in Pulse Width Modulation pins along with UART (Universal Asynchronous Receiver-Transmitter) communication. It is also able to provide SPI communication, which is Serial Peripheral Interface, which enables us to communicate with other modules. It also provides two built in power source pin. One provides 3.3V VCC and the other provides 5V VCC. It can easily be powered by 12V DC. We can program it using the Arduino Integrated Development Environment using C language.

SIM800L is a highly integrated GSM/GPRS module. It is primarily designed for mobile communication. It is developed by SIMCom. SIM800L is a budget friendly module which makes it a primary choice for anyone who wants to integrate basic communication in their project or product. It has the ability to act like a mini telephone as this supports both voice and data communications. You can send an SMS, receive an SMS, send and receive calls using this module. It generally is a very low power consumption module but there are a few technicalities which will be discussed in detail later in this chapter.

NEO-6M is another integral part of the system. It is a compact but highly accurate GPS (Global Positioning System) which is used for tracking location and other navigation application. Because this has a compact design, it is a forerunner for any IOT person who wants to integrate location tracking or GPS in their project. It uses satellites and other signal processing and communications algorithms to provide an accurate location. It is a very favourable module to use when working in remote locations. It can be paired with a microcontroller as it possesses the UART interface which helps us interact with the microcontroller. It also comes with a battery hence it is able to store the location for a little duration of time. Generally, this is a low power consuming device which makes it an ideal choice for the current project.

To power the whole system, we will be using Lithium Polymer batteries. They are widely used in consumer electronics because of their ability to recharge over and over again. They have a longer lifetime compared to other batteries and a comparatively minimal self-discharge rate. They need to be carefully used as they are hazardous and very reactive to certain things such as water. Further they are very sensitive to overcharging which could hurt the battery life over time. For charging these batteries we will use a TP4056 module. This module is specifically made to charge for lithium batteries. The main aim of this module is to regulate the charging process and protect the batteries from any mishap such as overcharging and short circuits. The input for this module is a standard 5V USB-C charger which is used for any other regular mobile device. We can choose different input ports for this module such as USB-C and USB Mini.

Obstacle Detection can be performed by a sensor, more specifically an Ultrasonic Sensor in our case. *There are many options, but they will be discussed later in detail*. Then we will use a buzzer to transmit the warning to the user whenever the Ultrasonic Sensors detect an obstacle. But we need a complete system to perform this, and we need components to coordinate with each other. The brain of the system is the Arduino Nano which comes with an AT mega 328 micoprocessor. The method mentioned above covers the basic requirement and is a very simple solution to the problem. But we can go a step further to improve this system by a mile. This could be done by adding to more modules, one which is a GPS Module, NEO-6M, it outputs the location using satellite communication and also a GSM Module, SIM-800L, which is used to connect to the internet in order to access the internet to send the data. Both of these modules work together via the Arduino Nano. They only work in one condition. Which is when the emergency or the SOS button is pressed. What this does is that it triggers the GPS Modules, which retrieves the current location. Then this location is forwarded to a pre-defined mobile phone number. The message contains the longitude and latitude of the location along with the Google Maps link. The pre-requisites of the whole system are that we need a battery to power the system but as we are using a lithium battery, it can be recharged. Furthermore, we need a GSM Sim Card so it can have the credit on it, which will enable it to send the SMS.

1.6 Software

The software will be a very straightforwardly coded because we are not working with any complex modules. We will try to have as much functions as we can so the code could look much better. Along with that the algorithm should be as follows. The system should scan the obstacles, if there is an obstacle it should turn the buzzer on. That is the normal state. Then whenever the button is pressed, the system goes into emergency state. In that state it will retrieve the GPS location which will then be formatted accordingly in the code. After it is formatted, it will be sent to the designated phone number using the SIM800L module. Simultaneously, there are LEDs included in this project. These LED's will start blinking once the SMS is sent. Along with that the ultrasonic sensors will be also keep on scanning any obstacles. The code will be written in C Language. Arduino Integrated Development Environment software will be used to write the code in and then the code will be flashed directly on the Arduino Nano Development board.

1.7 Competitiveness

As far as the economics of the project is concerned, we need to keep this as minimal as possible. Although, this really depends on the type of equipment we use. As we could get standard Ultrasonic Sensor cheaper, but better and accurate Ultrasonic Sensor for a much higher price. One of our aims was to keep the system as compact as possible because that does contribute to the economics as when the system is mass produced, it will affect the overall production cost. Hence, that was once of the reason to use an Arduino Nano in place of an Arduino Uno.

In addition to this, there are similar products in the market, but their prices are much exaggerated. These include WeWalk Smart Cane, priced at £385.00[4] and UltraCane which is priced at £625.00[5]. Therefore, this is also a driving force behind this project to make the system as much technologically advanced and efficient along with it being economically viable for a common consumer which will obviously encourage the consumer to buy it which will drastically make the movement of visually impaired people. In our opinion, this is one of the factors why these products are not popular in the consumer.

[4] (Sight and Sound Technology, 2024)
[5] (UltraCane, 2024)

Chapter 2 Background Research

Chapter-2: Background Research

2.1 Obstacle Detection Techniques and Systems

Obstacle Detection Techniques have been developed over the years since the early 1900's and the development has picked up pace in past decade as they have become popular due to increased focus on Autonomous Vehicles as well as Robots. These are the industries in which these techniques are more popular compared to any other industry in recent times. Obstacle Detection is the most important aspect of our project.

There are various techniques which are used to detect obstacle. Primarily, Sensors are preferred for this sort of task. But now, we have many new and advanced technologies. If we categorise the techniques, we have two prominent techniques. Firstly, Sensor based techniques and secondly, Vision based techniques. We do have subcategories for each of these but for now we will only discuss about the sensor based techniques. There are two types of sensors. Active Sensors and Passive Sensors[6].

Active Sensors are the sensors which don't have their own source of power, they are externally powered. In addition to this, they produce the energy on their own e.g. waves. Whilst Passive Sensors are the opposite of this. They are not powered externally. Infrared Sensor can be considered as a Passive Sensor if the Infrared Receiver is not being used in pair with the sensor.

2.1.1 LiDAR

There are many types of Active Sensors. These include LiDAR, Sonar, Radar and Ultrasonic Sensor. LiDAR stands for Light Detection and Range. It is a very high precision sensor compared to others. The reason behind this is that it emits short bursts of light. After it emits light, the light returns to the sensor, the time difference between emission and receiving is calculated and then it is processed to compute the distance[7]. But the issue with this is that the data is so huge that it takes time to process hence it slows down the decision making process. There are two types of LiDAR sensors, 2D and 3D[8]. 3D systems are much more complex, slower, and expensive than the 2D one. 2D gives us distance information whereas using 3D we can retrieve the height of the object. LiDAR are very popular amongst the sensors used in the autonomous vehicles industry and robots.

[6] (Yadwinder Singh, 2017)
[7] (Tufts University, 2022)
[8] (Ramos, 2016)

Figure 1, LiDar Sensor by DfRobot[9]

2.1.2 RADAR

RADAR system is another Active Sensing system. RADAR stands for Radio Detection and Ranging. Its functionality is very similar to LiDAR's functionality. RADAR uses electromagnetic waves between the frequency of 220Mhz and 35Ghz which are transmitted using a power amplifier and an antenna.[10] The reflected signal is then processed through multiple processes similar to an RF signal. After processing, we are able to determine the position of the object, traditionally using Cathode Ray Tube. It was developed back in the World War 2. But it has a variety of applications if we compare it with LiDAR. It is used in Air Traffic Control, Ships, Aircraft anti-collision system etc.. An advantage of Radar is that it can be used in any sort of condition, day or night. Along with that all weather conditions are suitable for Radar[11].

Figure 2, Radar Sensor[12]

2.1.3 Ultrasonic Sensor

Similar to the previous two techniques, Ultrasonic Sensing is another technique. It uses ultrasonic or sound waves to detect the objects. The frequency of these waves is generally higher than 20kHz[13].

---

[9] (Farnell, 2024)
[10] (Bhatta, 2017)
[11] (Discant, 2007)
[12] (Farnell, 2024)
[13] (L. Koval, 2016)

The sensor measures the time between transmitting and receiving the sound wave and then with the help of speed of sound we can calculate the distance.

$$d = \frac{c.t}{2}$$

Equation 1, Measuring Distance in Ultrasonic Sensors[14]

The transducer generates the signal which is then propagated as waves. In some sensors it can also operate over a range of frequencies. Primarily it is a distance measuring technique but there is another use of it which is SONAR, it stands for Sound Navigation and Ranging, which uses ultrasonic waves under the water. It works well underwater as sound waves travel farther in the water compared to waves of RADAR and LiDAR. The issue with this technology is that the sound waves are dependent on external factors which include weather and the medium they are travelling in as it can change the speed of sound waves transmitted. But these waves a prone to things like Dust and Light. A flexibility we have in this technology as the waves which is generated, comes out as a cone shape hence we can change the angle of the wave to vary the range of distance measurement.

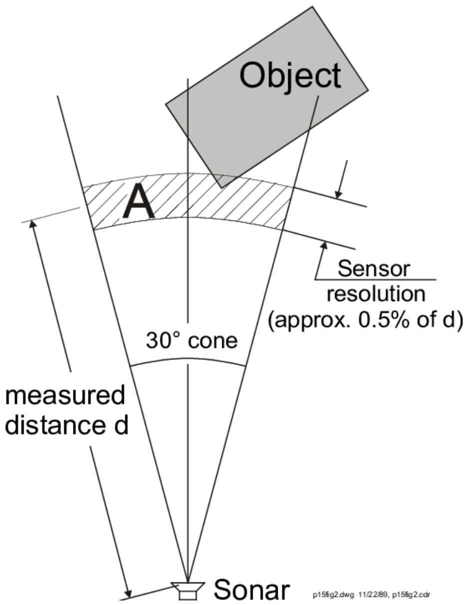


Figure 3, Ultrasonic Wave Propagation[15]

If the angle is small, we can measure a large distance and vice versa. But compared to other sensors the precision does vary depending on the application you are using it for. As SONAR system might be using different frequencies compared to an Ultrasonic sensor. Usually, a single Ultrasonic sensor operates on a single frequency, in our case we use HC-SR04. It operates around 40Khz but measure around 10m distance. In this technique we are only able to measure in 2D.

[14] (Tarulescu, 2012)
[15] (Shoval, 2001)

Figure 4, Ultrasonic Sensor HC-SR04[16]

Infrared Sensing is another technique which is commonly used in obstacle detection and distance measurement. Infrared Sensor acts as an Active and Passive Sensor which totally depends on the way you use. If you are using it in pair with an Infrared Receiver, the photodiode, then it is an Active Sensor like Ultrasonic sensor but in its single capacity it acts as a passive sensor. It also emits waves, which are infrared waves at certain wavelength. In the infrared spectrum, all objects radiate some sort of thermal radiation which is also detected. Although this is only possible if it is used as a passive sensor.

2.2 Analysis

We have discussed a wide range of obstacle detection techniques. Each of them have it's own pros and cons. But firstly, we need to see what our requirement is. We need something compact and light. We need something economically feasible along with it being precise and easy to use. If we start from LiDAR, LiDAR is the most accurate and precise system we discussed above. Recently, time of flight LiDAR are in the market which can be used for IOT purposes. They are cheap, compact and light weight. One these sensors is the TOF10120[17]. It is a very cheap option to go with. It takes in almost 35mA of current. But the Ranging Range mentioned in the datasheet is 100-1800mm. So this is key factor why we won't be using this sensor as we want to give the user maximum time to get alerted in case there is an object which means we need to have a sensor which has a larger range of ranging. But we can go for a higher range LiDAR sensor but they aren't suitable for IOT applications and project like ours moreover they are much more expensive.

Similarly, like the TOF10120, we have IOT radar sensor available in the market. One of them is the SEN0395[18]. This sensor requires 90mA current. It operates at 24GHz frequency. Further it has a detection range of 9M. The sensor in itself is very portable, compact and absolutely suitable for our application. But 90mA is a lot of current in our situation as we will be using multiple components. Furthermore, it is priced at £27.40 which is expensive.

Next are the Infrared Sensors. For Infrared Sensing, firstly we need to different components, the IR transmitter and IR receiver. Although these two come in a single module but the inherently are more suitable for a consistent environment as the detection range could be effected by the obstacle colour. The range of detection is between 2cm to 30cm[19]. Also it is prone to Sunshine effect as the sunshine

---

[16] (Farnell, 2024)
[17] (Electronic Clinic, n.d.)
[18] (Farnell, 2024)
[19] (ThePiHut, 2024)

does contain waves which come into IR spectrum and could disturb the output hence it will not be suitable for use outdoors.

Then we have Ultrasonic Sensors. They are very popular in the IOT space around the world. There has a lot of work been already done using these sensors so it is much easier to work with such components. The sensor of choice in such a scenario is definitely HC-SR04[20]. These are Ultrasonic Sensors in Monocular Configurations which means it has two transducers. Furthermore, the current required is almost 15mA which is very significantly lower than the sensors discussed prior to the this. Along with that it has a max measurement range of 4 meters. Which in our opinion is a very suitable range. These are slightly larger compared to other sensors but the obvious reason for that is the monocular configuration. Hence this makes it the best choice for being used in our project.

Another thing to add here is that HC-SR04 is a very popular sensor and has a widespread use. Hence it is much easier to find resource to code it compared to any other sensor. This also makes it is to connect work with other components as sometimes these might have compatibility issues.

In addition to this, LiDAR and Radar are much more complex technologies than Ultrasonic. The data processing algorithm surely are different but we should take this in account that they definitely would effect the response time of the system hence ultrasonic sensor is a much better choice.

The following table brings more insights into the comparison we just discussed.

	HC-SR04	TOF10120	SEN0395
Max Range(m)	4	1.8	9
Current(mA)	15	35	90
Dimensions	45*20*15mm	10*20*13mm	24*28mm
Cost	US$4.50	NA	£27.40

Table 1,Comparison of Ultrasonic Sensor, LiDAR and RADAR

2.3 GPS

GPS is an abbreviation of a broader term, Global Positioning System[21]. It is a common tool in today's world to track a location of any device or even anything in general. In summary, this system uses satellites to track the location of the object. What it makes it more interesting is that this technology doesn't require internet hence it can be used offline, which makes it favourable to use in remote locations as well.

The usual process involves two devices, the satellite, and the GPS device which we have. The GPS Satellites are broadcasting radio signal around the Earth as there are 31[22] of these currently in space. The device needs signal from at least 3 satellites to workout the exact location as there is calculation required to precisely locate the object. These satellites have atomic clocks[23] on board to make the calculation as accurate as possible. The data which is sent to the receiver contains data from the 3 satellites as well as the time from atomic clock. The receiver has to be setup in such a way that it is firstly able to decode that information from the signal. Secondly, it should also account for any delays in the transmission of the signal. Although that the signal speed is almost equal to the speed of light but as there is distance between the receiver and the satellite, it takes time for the signal to travel. In

[20] (Sparkfun, 2024)
[21] (U.S Department of Defense, 2018)
[22] (Federal Aviation Administration, 2022)
[23] (GPS.GOV, 2022)

addition to this, the signal has to travel the different layers of the Earth like ionosphere, stratosphere etc. which also accounts in the propagation delay of the signal. Lastly, the atomic clocks being in the space are 38 Microseconds[24] ahead of the Quartz clock on Earth hence this is also taken in account so that the location is precise. The calculations are done with the help of geometry formed by the 3 satellites and the time from the atomic clocks from the satellites. But using a 4th satellite, it removes the need of atomic clocks[25]. This also enables the receiver to calculate the 4 important thing which are required to determine the location, which include time, altitude, latitude and longitude.

A usual GPS is very precise while locating you. It is accurate up to 7.0 meters for 95% of the time.[26] Along with that it is free to use for anyone. It is used different fields including Military, Surveying, and Navigation.

There are two technical principles behind GPS, Trilateration and Synchronization. Trilateration is a technique which is used to determine the distance from known points[27].


Figure 5, Trilateration, Spheres created by GPS Satellites

The GPS receiver uses the distance from multiple satellites using the signal using the calculation of the time it takes for the signal to reach to the receiver. Hence in result of these calculations we are able to get the longitude and latitude. Trilateration could 2-D and 3-D. In 3-D we are able to find the altitude of the receiver as well.

We earlier mentioned that there is a slight time difference between the atomic clocks on the satellite compared to the clocks in the GPS receiver on Earth. The reason behind this is that the gravitational force in the satellite is weak compared to that on the Earth[28]. Hence as a result of this the clocks in the satellite apparently run faster compared to the clocks on Earth. Although the clock on the satellite is in motion, which makes it slower due to time dilation as told by Albert Einstein, but the resultant difference is of almost 38 microseconds per day.

The reason to discuss GPS is that we will be using a module, NEO-6M, which makes the use of this technology hence it was necessary to discuss this technology beforehand.

2.4 GSM

GSM is a landmark achievement in the field of communication. GSM stands for Global System for Mobiles[29]. This is one the key technology that enabled the 2G communication[30] in the early 1990's. It was a giant leap from 1G which was an analogue communication system. It enabled people to send

[24] (PennState College of Earth and Mineral Sciences, 2024)
[25] (Federal Aviation Administration, 2022)
[26] (Federal Aviation Administration, 2022)
[27] (GISGeography, 2024)
[28] (PennState College of Earth and Mineral Sciences, 2024)
[29] (TMobile, 2024)
[30] (Ofcom, 2015)

SMS messages, call each other but with much more clarity and also access internet although the internet speed was not that good[31].

One of the reasons why it was adopted quickly and became popular worldwide was it's ability to operate in different countries and compatibility with other network providers. This enabled a feature called "roaming" for normal people[32]. Consequently, it made travel much easier and international connectivity was on doorsteps.

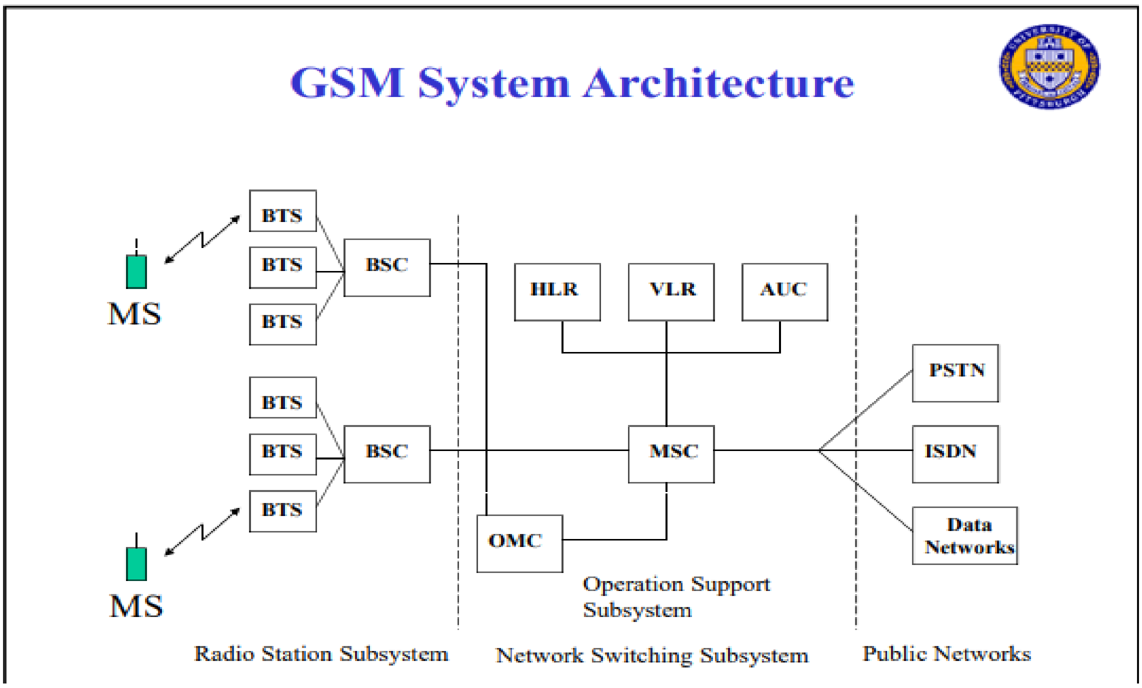


Figure 6, GSM System Architecture[33]

The architecture of GSM generally consists of a couple of main stages as shown in the diagram above. The mobile is connected to a Base Transceiver Station (BTS).[34] The medium they are connected with each other are air as the BTS propagates radio signals into the air which are then picked up by our mobile phones and other GSM devices. Base transceiver station is further connected to a Base Station Controller (BSC). These two are connected together by Abis Interface. A single BSC has capability to control multiple Base Transceiver Stations hence it is a usual practice that a BSC controls multiple BTS's. BTS and BSC combined make a subsystem in the GSM architecture which is named as Base Station Subsystem. This BSC is now connected to a greater system which is called the Mobile Switching Centre (MSC). This is MSC interconnected to multiple separate modules which include Home Location Register (HLR), Visitor Location Register (VLR), Authentication Centre (AUC) and Equipment Identity Register (EIR)[35]. BSC and MSC are connected to each other via an "A Interface". There is an additional module which is used at the customer end, which is called Subscriber Identification Module (SIM). This connects us to this whole system via BTS initially. AUC is an important module among those separate module. This is of immense significance because this module

[31] (Ofcom, 2015)
[32] (University of Pittsburgh, 2011)
[33] (University of Pittsburgh, 2011)
[34] (Aircom International, 2002)
[35] (The International Engineering Consortium, 2024)

makes sure that the caller is an authentic person. This is done by accessing the security key in the SIM of the user. Along with this, during the communication, an encryption algorithm is employed to enhance the security. The MSC is also further connected to a Public Switched Telephone Network (PSTN)[36] which is responsible for switching the calls and directing these calls to the right person.

The core of the GSM technology is the TDMA and FDMA[37]. These are Radio Multiple Access technologies, Time Division Multiple Access, and Frequency Division Multiple Access respectively. They function on different principles compared to each other. In TDMA, the data for one user, is not transmitted continuously. As multiple people are accessing the spectrum, the transmission is queued hence the whole spectrum is used but, in a queue because the channel is split into sequential time slots, so multiple people are able to access the spectrum for a certain time frame during the assigned time slot. If we compare this to FDMA, FDMA has divided the spectrum into separate frequency bands. So, a user is assigned its own unique frequency band. The user is able to transmit and receive data simultaneously in this technology.

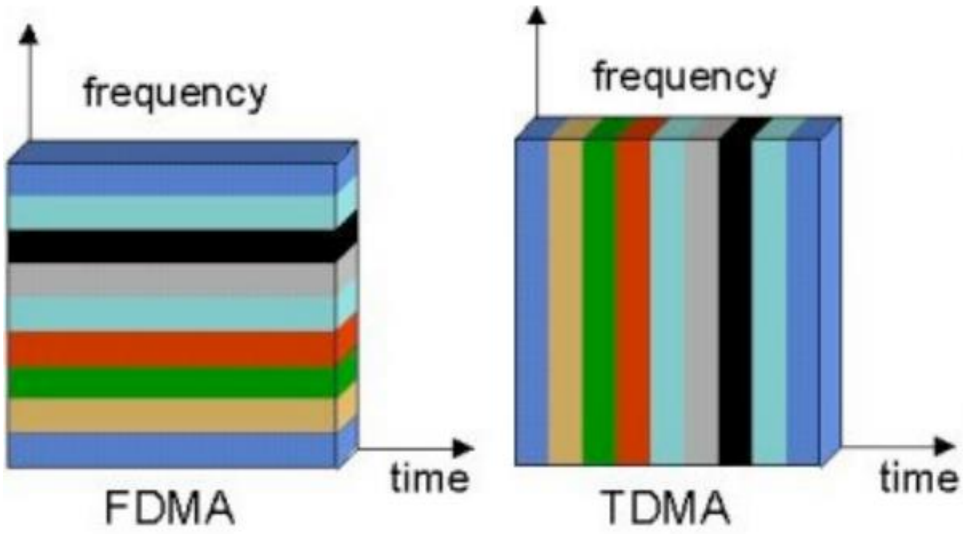


Figure 7, FDMA and TDMA comparison[38]

If we do a straightforward comparison, these technologies have their own pros and cons. TDMA is more demand oriented. It can adjust itself according to the demand and prove to be flexible by allocating the timeslots efficiently[39]. But in FDMA, demand is not a factor for the resource to be allocated. Once the frequency band is dedicated to a user, it can't be given to any other user hence a limited number of people can use the spectrum when FDMA is being used due to non-availability of frequency band. But TDMA on the other hand has very complex synchronization and timing mechanisms whereas FDMA is much easier to work with as there is less handling of data due to less number of people using it compared to TDMA.

It was highly important that we discuss how GSM works because SIM-800L module, which we are using in our project, its core is the GSM technology hence in our opinion it was necessary to elaborate the technology and architecture of GSM.

---

[36] (University of California, Los Angelos, 2024)
[37] (Aircom International, 2002)
[38] (University of Roma, 2008)
[39] (GeeksforGeeks, 2020)

University of South Wales Prifysgol De Cymru

2.5 Product and Literature Review

There are many works and researches which have been done Smart Stick for Visually Impaired people. We have gone through a few of them. The key difference between their approach and ours is that we want to create something, despite being similar, it should be feasible from economic point of view. We will talk about economic feasibility further as we go on in this report as there is not point in creating something which is not economically feasible. And all these projects do lack this perspective.

WeWalk:

This company is experienced in developed technologies for the visually impaired. They have been in business for more than 10 years[40]. In 2017, they launched their flagship product which is the white cane by WeWalk. It is a smart stick for the visually impaired. They have a proper team which works on these projects further they also do Research and Development in this field of technology.

Talking about the smart stick they offer; it is priced at £539. The main feature of any smart stick for blind people would be detecting obstacles. They use Ultrasonic Sensors to detect any obstacles. The detection is sent to the user as feedback using Vibration Motors. The USP of this device is the touchpad along with the smartphone app which is provided to the user. Using the touchpad[41], you can perform many functions without having to use the mobile phone. With this you can navigate and get directions effectively as it provides turn-by-turn instruction for the journey to the user. It also assists the user to help find different places such as restaurants and different landmarks. It goes a step further in which they have introduced a feature which will allow the user to get access to the transit information[42]. This means that access to public transit will be made easy for the visually impaired people as the users will be able to get directions to the transit as well as they will be able to get access to the timetables.

In addition to all of this, as they have a whole team[43] which works behind this product hence they are able to provide regular software updates which can improve the user experience continuously by fixing different bugs and along with that introducing new features.

In our opinion, this product is a very well thought product and provides very necessary options for any visually impaired person. But there are a few thoughts which are important to be discussed. Firstly, being priced at £539 is a little over the top in our opinion to what they are providing but we still think that this should be relatively lower. The reason for this is that not every visually impaired person would be able to afford this hence it feels like a luxury. It may be affordable in first world countries but for developing or underdeveloped nations this definitely is considered a luxury.

Secondly, they are very dependent on smartphone to provide directions to places. There is no mechanism in place, apparently, in case the smartphone's battery finishes. This can lead to a confusion and a panic situation for the user.

Lastly, Vibration Motors might be irritating for many users. Something consistently vibrating in your hand would make you let the thing go.

[40] (WeWALK, 2020)
[41] (WeWALK, 2020)
[42] (WeWALK, 2020)
[43] (WeWALK, 2020)

UltraCane:

UltraCane is a product by Sound Foresight Technology Ltd. It is a standard cane which uses narrow beam technology instead of sensors to detect obstacles. This company has been developing mobility aid since 2010[44]. This product is priced at £625. It claims that it detects obstacles within 2 or 4 meters as these are the only 2 settings provided by the company. Just like WeWalk it uses two vibtrating buttons which are used to give tactile feedback to the user. These buttons produce a varying range of frequency depending on the distance from the object. These buttons are also able to tell the user about the direction of the obstacle.

UltraCane is a very decent product for standard use. Although being priced at £625, this product lack many key features which are very easy to develop and integrate in this period of time. The most important thing is that this product lacks the idea of emergency state as just like the WeWalk smart stick. It lacks almost all the features of WeWalk smart stick except the obstacle detection part. UltraCane is inefficient compared to WeWalk. Furthermore, many other research projects are comparatively better in terms of feature which are done in this field. In addition to this, the important thing is that it is priced at £625[45]. We think that our project offers much more despite not being commercialized and our project is relatively much more affordable than this product. They could have added many other functionalities such a location tracking, an emergency state, or any sort of communication methods in case of emergency. Along with that their website looks really old, and we doubt it has not been updated since a long time. This also is an indication about their poor support for this product which clearly means that the company hasn't been updating the product over the years which has been an obstacle in adoption of modern technology.

Paper 1: Smart Stick for the Blind and Visually Impaired People[46]

This project is a standard Smart Stick fulfils all the theoretical requirements of a Smart Stick for the blind people. Its obstacle detection is done using the Ultrasonic Sensors although they have not provided any reason why they are using this specific sensor as there other sensors available. In addition to this they are using a water sensor to detect if there is any puddle hence the user is able to avoid the puddle. An RF Transmitter and Receiver are also used. The purpose of these is that when the stick is lost or misplaced, using this form of communication, along with a buzzer, the user can find the stick. Other than that, it uses Sim 808 Module and a GPS Module. They have mentioned that the reason to use this is because 2 way GPS communication is expensive hence this way it is much easier to communicate with satellite. This sim module also lets them send messages to the predefined mobile number when the emergency button is pressed. To indicate the user if there is an obstacle or not, they are using vibrating modules. There frequency of vibration changes depending on if there is an obstacle or not. The most interesting part of their project is that they are using Texas Instruments MSP430G2553 as their brain of the system.

This generally is a really good system for the visually impaired. It provides all the essential features which we have provided in our project as well. But they have used different components. In our opinion, what they have done could have been achieved using different, simpler and cheaper components as well. This system is over powerful, which makes it in efficient. Thus, we can also assume that this system will be costly as well. Components such as Sim808 module attract attention as this is an upgraded version of SIM800L. SIM808 supports 4G hence it opens up much more avenues

[44] (Ultracane, n.d.)
[45] (Ultracane, n.d.)
[46] (Gupta, 2018)

of technology such as Live Location Tracking. This paper does not discuss this, and it lacks the clarity. In addition to this, MSP430G2553 is not a go to choose for IOT projects because it lacks community and support in case something goes wrong, it might be hard to figure out.

Paper 2: Smart Walking Stick for Visually Impaired People Using Ultrasonic Sensors and Arduino[47]

This paper is aimed at creating a smart stick for the people who are visually impaired but using minimal components as they have mentioned using Ultrasonic Sensors and Arduino. They are using Arduino Uno as the brain of the system. They have interfaced buzzers which buzz once, twice, and thrice depending on the detection. Either it was an obstacle or water. To detect obstacles, they are using Ultrasonic Sensors and for the water they are using water sensor.

This paper is very generic display of a Smart Stick for the blind people. It fulfils the general goals which are obstacle detection. But there are a few places where this project has the potential to improve. Firstly, the picture shown in the paper display that the ultrasonic sensor is mounted just by the handle on the top of the cane. This raises serious concerns regarding the ability of the sensor to detect obstacles in general as the positioning of the sensor in our opinion could have been different. Furthermore, the same picture shows us the case which apparently holds the whole system. All the components are packed in one compartment together. Here our concern is that this might lead to overheating issues which affect the working of the system. As we have to make sure that this system works in all sorts of weather conditions.

In addition to this, this project uses Arduino UNO along with one ultrasonic sensor, water sensor, LCD, and a switch. In our humble opinion, instead of Arduino UNO some other development board could have been used for example, Arduino Nano or Arduino Micro. This makes the system inefficient as it is overpowered furthermore the form factor of Arduino UNO is large if compared to Arduino Nano or Micro. Along with that, for a fact that Arduino UNO has same number of digital input and output pins as an Arduino Nano hence it could save a lot of money, space and technically make the system much efficient. There has been no discussion over the economics of the whole system but considering the chosen components, it should be relatively cheaper, but we should keep in mind that this does the basic obstacle detection only with flaws in the design.

Paper 3: Multi-Functional Blind Stick for Visually Impaired People[48]

This project aims to detect obstacles, staircases, and wet surfaces. They are using a combination of Infrared Sensors along with Ultrasonic Sensor. The IR sensor detects any stairs on the bottom to detect stairs. 2 Ultrasonic Sensors are used to detect obstacles. One is mounted at the bottom other is 2/3 from the bottom. The output to the user is given via the speaker module or the vibration motor. It has pre-recorded audio responses hence depending on the situation it outputs the specified audio message. It also contains an emergency button, which with the help of GPS, GSM module is able to send the location coordinates of the user to a predetermined mobile phone number. They also include a find and lost system which they establish using RF Transmitter and receiver. When the stick is misplaced, the user can use the rf transmitter remote to trigger the buzzer which is mounted on the stick. The prototype is also shared in the paper, which is made out a PVC and it is all covered with wires and components from top to bottom. Further in the results they show that the stick works perfectly. To modify the predetermined phone number, a GUI app is developed as well through the mobile number can be changed.

---

[47] (Gbenga, 2017)
[48] (V. Kunta, 2020)

The paper reflects a very thoughtful demonstration of a smart stick. But it lacks the justification for why certain components are used. There has been no discussion about any other alternatives. The literature review done in the paper is not detailed. In addition to this, the flowchart of the software is missing hence it is not possible to interpret the working of the software instead they provide a general flow chart of the working mechanism. Besides this, this project does not have any compartment or a box to hold all the components. Instead, the components are all spread over the stick vertically. The picture which is displayed in the paper gives the impression that a PVC pipe of a thicker diameter is used to act like a stick. This is concerning as this will really affect the ergonomic ability of the product and also reduce the ease of use of the product. It uses a SIM900A module, which is an advanced module compared to the SIM800L. In our opinion, there is a lot of potential of creativity if SIM900A is used such as live tracking. Hence, addressing these issues could significantly enhance the overall quality and functionality of the smart stick project.

2.6 System Design

The main task which is crucial in making this idea work is that the design of the system should be robust and mobile. A robust system should be versatile enough to survive in different situations. For example, different climate and weather conditions. In our scenario, visually impaired people exist around the globe, so we need to implement such solutions which are globally viable. In addition to this, if the systems is huge in terms of size and heavy in terms of weight, that is a big problem especially for the problem we are addressing as this could be directly affect the physicality of the user over a period of time. Along with this, if the system is huge and heavy, the manufacturing cost would definitely, be high due to the footprint of the system. Furthermore, if the same thing could be done by using a smaller system, then it would an inefficient system along with it being an expensive system. Another nuisance of a huge system is the higher power requirement. Hence, we need to keep the system as small as possible especially for this project. As it would improve ease of use for the user. Along with this it will be economically affordable as well due to the reduced manufacturing and component costs.

This is the ideology we have followed and tried to work to fulfil the requirements of the project.

2.7 Components

Coming to components now, the components are very carefully selected due to a lot of complications due to newer technologies which sometimes are not compatible with the technology which is older. Also understanding the newer technology can be very hard as there is not so much stuff and resource to work with. Hence, our main goal was to make sure every component is compatible with one another. We will now list the components and discuss these details as the components are highly interdependent.

We talked about the system being mobile earlier and how important this is hence we chose the Arduino Nano as the brain of the system. Arduino Nano is a small development board with a microprocessor. It has a small form factor which makes it much suitable for smaller IOT projects. It has 22 digital input and output pins.[49] This absolutely complements its small form factor hence making it the perfect choice for our project. As we will not be using many different modules. In addition to this, there has been a lot of public work already done in terms of libraries for the components, so it is easy to work with most of the components.

[49] (Arduino, 2024)

There are other boards of similar qualities available which include, Arduino Micro and Arduino Mini. Even Arduino Nano has different variants which include additional functionality. The following table displays the summary of variation of specification of different Arduinos.

	Arduino Nano	Arduino Nano 33 BLE[50]	Arduino Nano 33 BLE Sense[51]	Arduino Nano 33 IoT[52]
Microcontroller	ATmega328	nRF52840	nRF52840	AMD21 Cortex®-M0
DC Current per I/O Pins	20 mA (I/O Pins)	15 mA	15 mA	7 mA
Input Voltage	7-12V	21V (max)	21V (max)	21V (max)
Digital I/O Pins	22	14	14	14
Power Consumption	19mA			
PCB Size	18 x 45 mm	18 x 45 mm	18 x 45 mm	18 x 45 mm
Weight	7 g	5g	5g	5g
Price	€21,60	€22,80	€35,10	€22,80
		Bluetooth® and Bluetooth® Low Energy		

Table 2, Comparison of different version Arduinos

In the table above we can see the different features of the different Arduino Nanos which are available in the market. If we compare the rest of the Arduino Nano's to the standard Arduino Nano, the only main difference is that they have a different microprocessor. The reason definitely is that the other provide different functionality. Like BLE is able to provide Bluetooth and Bluetooth Low Energy and it comes with certain sensors. BLE Sense has additional sensors compare to standard BLE. If we compare them to Arduino Nano IOT, IOT features Wi-Fi connectivity and is perfect for small IOT projects as IOT projects must have a network to communicate over.

There is a key difference between the standard Arduino Nano and the other which made us choose Arduino Nano. It is the number of Input and Output Pin. We wanted something which was small but did all the work which we required. Hence, it was the perfect choice for us. In addition to this, the standard Arduino Nano is slightly cheaper as well compared to the rest of them. We never intended to use the extra additional features which are offered in the other Arduino Nano's hence it would have been a wrong decision to select the other Nano's in the first place.

We also had the option to choose Arduino Uno. The first and foremost reason why we didn't choose Arduino Uno was that it has a much larger form factor compared to Arduino Nano. Although it has the same number of digital I/O pins compared to the Nano.

[50] (Arduino, 2024)
[51] (Arduino, 2024)
[52] (Arduino, 2024)

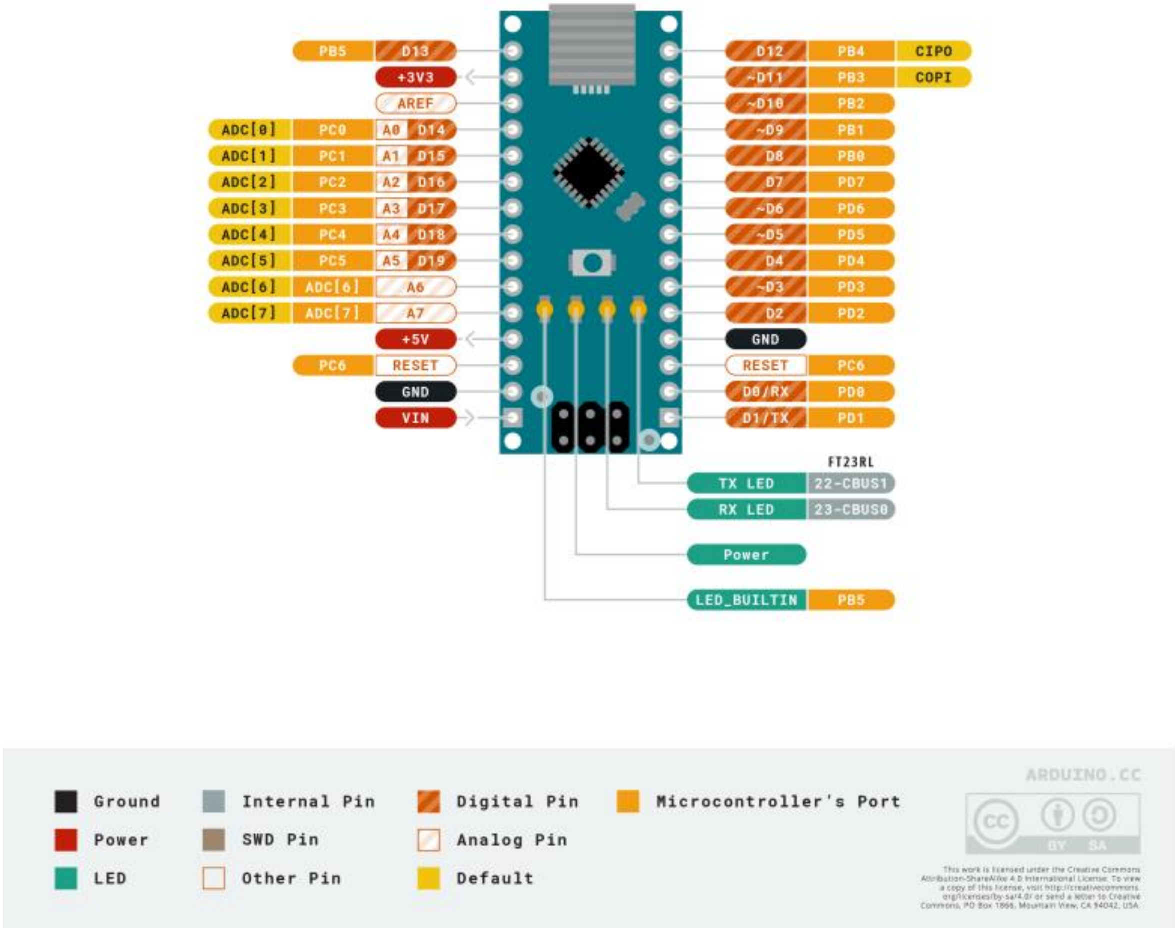


Figure 8, Arduino Nano Pinout[53]

The diagram above shows the Arduino Nano. On the right side of the picture, we have most of the digital I/O pins which we will be using for our project. On the left side, we have a couple of power pins along with Analog Input Output pins. A very interesting feature of Arduino is that we are able to power our component simultaneously if we have one which works on 3.3V and one on 5V. We can power multiple components like this. Also, we have to Ground pins which are interconnected. The VIN pin takes the input voltage to power this Nano. The typical value is between 7 and 12V DC[54].

The digital pins on Arduino Nano communicate with the microprocessor by using Serial Communication. The main role of the digital pins is to facilitate the communication between the sensors, modules etc. and the microprocessor. The communication is performed Serially. Serial Communication is a way of communicating digitally in which one bit of data is transmitted at one instance of time. This type of communication is enabled by the Universal Asynchronous Receiver/Transmitter (UART) infrastructure built in the microprocessor[55]. This converts the parallel data from the microprocessor to serial data for transmission. We have different commands in the Arduino IDE which tells a lot about how the communication works. To initiate the serial communication, we use the command Serial.being(baud_rate) using the software serial library. The reason of using software serial library is because there are only two hardware serial pins, D0 and D1.

[53] (Arduino, 2024)
[54] (Arduino, 2024)
[55] (Arduino, 2024)

Rest of the digital pins can be emulated for serial communication using the software serial library. Coming back to Baud Rate, it is basically the rate at which we are communicating serially in bits[56]. Usually, 9600 baud rate is used. Then we have Serial.available(), this command checks if the component is available to communicate. Serial.print(), Serial.write(), are used to print data in the serial monitor. Serial Monitor is a sub module in the Arduino IDE which is used to display the data which the Arduino is outputting in relation to our code.

But there is an issue here which is that the components can't communicate serially simultaneously. The reason lays in the software code. We need to end the serial communication from one component so that another component can start communicating serially. So for that firstly we need to enable serial communication for that specific component e.g gps.being(9600) and then end it by using gps.end().

2.8 Ultrasonic Sensor:

Ultrasonic Sensors are the heart of our project. Ultrasonic Sensors in our project are used to detect obstacles. We are using 2 Ultrasonic Sensors so we can have a maximum range of detection. We have already talked about Ultrasonic Sensors when describing different technologies which are used for obstacle detection hence you can click here[57] to see the in detail working of these sensors. HC-SR04 are one of the most standard but efficiently working sensors available in the market. These sensors operate at 5V DC[58] hence they can easily be powered by the 5V from the Arduino Nano. Their operating current is typically 15mA hence making them power efficient.

They work on a very basic principle of wave reflection. The sensor sends a pulse and then it waits to receive the reflected waves. When they are received it calculates the distance by measuring the difference in time. To be precise for HC-SR04 specifically, initially Trig pin is sent to 5V for 10us which then sends out a burst of 8 40kHZ waves[59]. Upon detection of the reflected waves, the Echo pin is set to high. The distance then can be measured by taking the width of the echo wave as time and use the following formulas to calculate the distance.

Time = Width of Echo pulse, in uS (microsecond)

- Distance in centimetres = Time / 58
- Distance in inches = Time / 148[60]

The whole phenomenon described above can be graphically scene in the following timing diagram.

[56] (Arduino, 2024)
[57] 3
[58] (Multicomp, 2024)
[59]
[60] (Cytron Technologies, 2013)

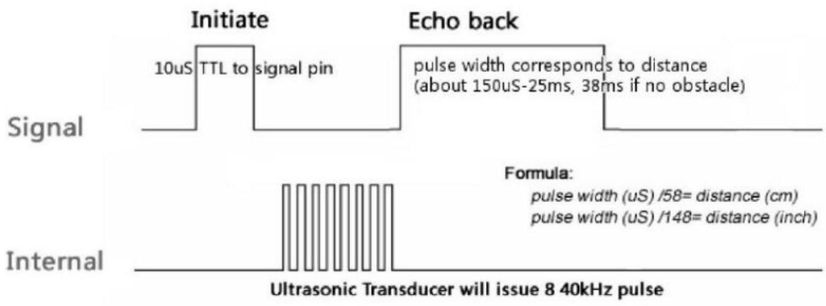


Figure 9, Ultrasonic Sonic sensor timing diagram[61]

2.9 GSM Module:

GSM Module is another distinguishing part of our project. It makes our project stand out from other projects. We are using SIM800L as our GSM Module. We have described what GSM is and how it works earlier in this document here[62]. But now we will specifically talk about the technicalities of SIM800L. SIM800L works at a range of voltage between 3.4V and 4.4V. But the recommended voltage is 4.0V. Arduino Nano can't provide greater than 3.3V and less than 5V at a single power pin hence we have to use an external source of power supply which are our Lithium Polymer batteries. But according to the datasheet it requires a power supply which can supply a burst of current at 2A. This burst of 2A is only for 577microseconds[63]. This is due to the high need of radio frequency transmission frequency. Hence, it is very important to provide a stable power source which can consistently be ready to provide 2A.

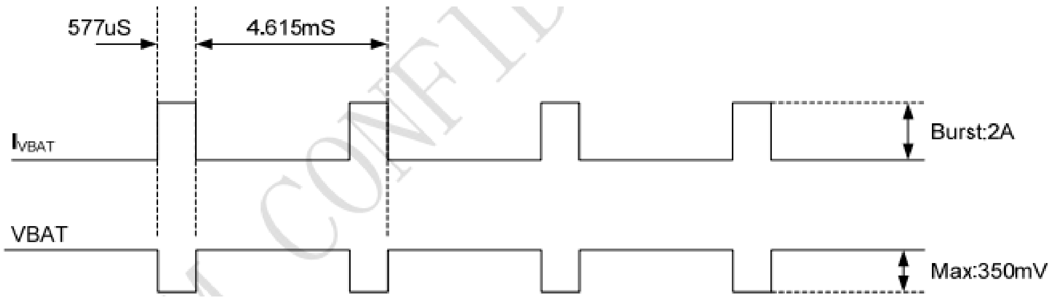


Figure 10, Burst of current during transmission[64]

It requires a SIM card to work. But the requirement here is that the SIM card should be 2G. This module does not support any other sim card of newer generation like 3G or 4G. This can connect to internet as well using 2G. Unfortunately, in the UK, 2G is being phased out gradually. Many operators don't offer 2G in a lot of areas, even if they do, none of them offers 2G Data services anymore as 2G is considered as a backup. We could have gone for newer versions of this module which support 3G and 4G like SIM79600CE but they are much more expensive compared to this.

[61] (Cytron Technologies, 2013)
[62] 2.4 GSM
[63] (SIMCom, 2013)
[64] (SIMCom, 2013)

Other than this, this module as a whole offer much more than we are using it for. This can make calls and receive them as well. Similarly, it can send and receive SMS as well. It has the ability to drive a speaker. Along with that it can connect to a microphone as well.

Figure 11, SIM800L GSM Module[65]

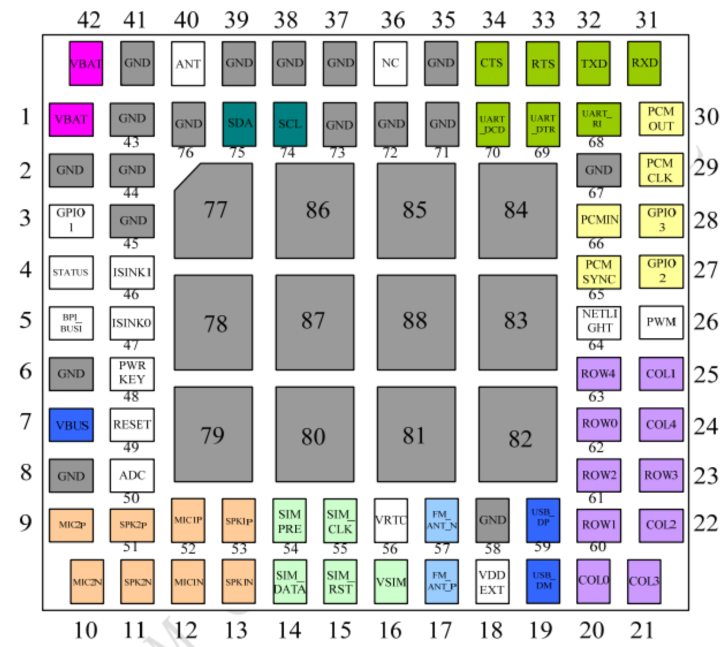


Figure 12, SIM800L Pinout[66]

The diagram above shows the pinout of this module. On the left hand side, we have the Power pins along with the transmission pins respectively. It also has a socket on the top left corner for an antenna. This antenna helps this module for signal reception and connectivity to the network. On the right-hand side of this module, we have ports for microphone, speaker, data pin and ring indicator. We will only be using the pins on the left-hand side of the module. This module comes with built in different

[65] (Components 101, 2021)
[66] (SIMCom, 2013)

operating modes[67]. There is a normal operation mode, power down mode and minimum functionality mode. We usually will be operating in Normal Operation mode, but it has sub modes as well. We will usually be operating in GSM Idle mode. In this mode the software is active, and the module is registered with the GSM Network hence it is ready to communicate.

We can actually change these operating modes as per our need. We can do by using commands called AT Commands. These are commands which helps us communicate with the module using serial communication through the Arduino Nano. These are predefined commands which are provided in the datasheet[68] so we can use them to complete the required tasks.

2.11 GPS Module:

Moving forward, we are using another module which is our GPS module, the NEO-6M. This module's fundamental function is to retrieve the current location. But there is a lot of technological procedure going just to get your current location using GPS technology. We have already talked about GPS technology before in this dissertation here[69]. But we will delve further into the technology for this particular module. The maximum voltage rating for this module is 3.6V and its current requirement is almost 100mA[70]. So, when we power it up for the first time, it will take a few minutes to start working properly. The reason behind is that it needs time to establish connection to satellite. And to that, first it needs to find if there is any satellite available or not. Practically, it can take a few minutes to get started. But as this module has different start type e.g. Cold Start, Warm start etc, the time to first fix, which is the time to capture the satellite signal and calculate the current position, TTFF, it is almost 27s for a cold start[71]. The reason why it has different start types is that this module comes with a small battery. Hence it stores the location data for some time, so it doesn't need to do the whole process which it does for the first time to connect to the satellite. To aid the module for connectivity, it comes with an active antenna as well.

Just like SIM800L, this also comes with different operating modes. It has three main operating modes[72]. The first one is Maximum Performance Mode, then Eco Mode and lastly Power Save Mode. In the Maximum Performance Mode, the module will continuously search for all the available satellites so it can obtain a position fix. When the position fix is obtained, the module will still keep on searching for satellites which were not tracked. The Eco Mode is technically the next stage after the Maximum Performance Mode. Before it keeps on additional satellites, but in Eco Mode, when there are sufficient number of satellites tracked the module will switch of its acquisition engine which will help conserve power. In the Power Save mode, the module will selectively turn on and off parts in the module which further reduce the power consumption.

There are other options available to use instead of the Neo-6M. But in our opinion, Neo-6M was the best option available for us. The reason behind this is that it is an economical module, and the form factor is of this module is small. Furthermore, this module has only 4 pins. One for input voltage, two for transmission and reception and one for ground as shown in the picture below.

---

[67] (SIMCom, 2013)
[68] (SIMCom, 2013)
[69] 2.3 GPS
[70] (u-blox, 2011)
[71] (u-blox, 2011)
[72] (u-blox, 2011)

Figure 13, NEO-6M GPS Module

It requires 3.6V to be powered. It is a very unique voltage as this can't be supplied by either of the Arduino pins. So, the getaway we figured out to solve this problem was that we can just power this with the 3.7V Lithium-Ion battery. We have practically tested this out and the module runs absolutely perfectly.

2.12 Lithium-Polymer Batteries:

Lithium-Ion Batteries are one of the most common components among all the modern day electronic devices. There is one primary reason which is that these batteries provide more capacity in a much smaller size compared to lead acid batteries. In addition to this they are rechargeable. Lithium-Ion batteries generally have an anode and a cathode. Along with that, they have an electrolyte solution. The anode in most cases is made of Graphite. The cathode is made up of metal oxides such as Cobalt Oxide or Manganese Oxide[73] etc. The electrolyte is usually a liquid in the case of Lithium-Ion batteries which helps the ions move from the cathode to the anode. This electrolyte is usually composing of lithium salt which is dissolved in an organic solvent. The lithium-ion batteries by default are a little rigid because they have electrolyte in them but because they provide a high capacity of charge, they are preferred in modern day electronic devices.

[73] (Massachusetts institute of technology, 2020)

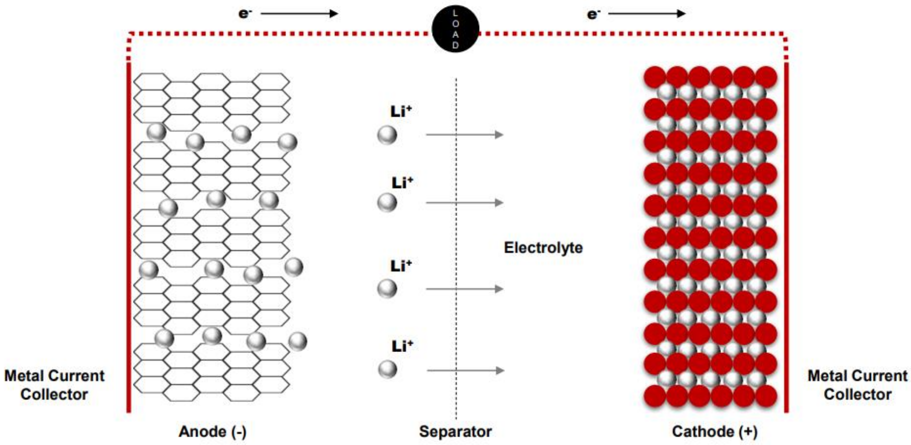


Figure 14, Lithium Ion Battery when charging[74]

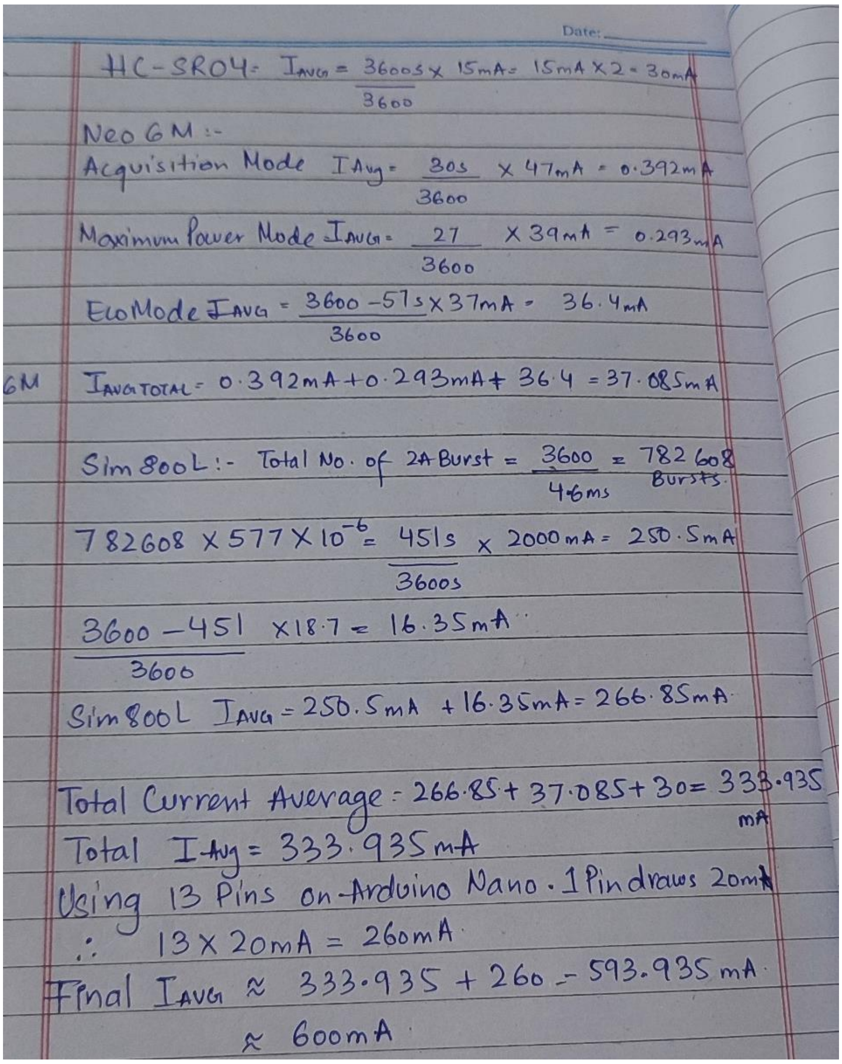
On the other hand, Lithium Polymer batteries are the same as Lithium-Ion batteries but there is only one difference. The electrolyte. In the Lithium Polymer batteries, the electrolyte is not a liquid instead is a solid like substance, more like a gel which facilitates the transfer of electrons between the anode and the cathode. This enables the polymer batteries to be much smaller compared to the lithium ion batteries. This also make them light weight and much more mobile. Along with this these are more suited for fast charging applications due to the lower internal resistance. Hence, this is the reason why lithium polymer batteries are preferred more for smaller electronics, hobby or IOT projects.

Generally, Lithium Ion batteries are not considered safer if we compare them to Lithium polymer batteries because L-ion batteries have a liquid electrolyte compared to LiPo batteries. This makes them more prone to thermal runaway. The reasons could be different like High Temperature, overcharging etc. .The solid electrolyte in LiPo make them relatively safer. This difference in electrolyte also enables LiPo batteries to be in different shapes and sizes. If we see from economical point of view LiPo batteries are comparatively expensive but in our opinion for the features and mobility they offer, they are worth the price. For our project we opted for two 3.7V 1500mah LiPo batteries.

[74] (NASA, 2019)

Figure 15, Lithium Polymer Battery for our project

We have used two of these batteries. Because we want to give the users a higher run time. So, by doing the calculation we first identified the current we will be using on average. In this we have not included Buzzer and LED's current because that is very minimal. Furthermore, this is an estimate of the average current. The calculations are shown in the image below.



HC-SR04: $I_{AVG} = \frac{3600s}{3600} \times 15mA = 15mA \times 2 = 30mA$

Neo 6M :-

Acquisition Mode $I_{Avg} = \frac{30s}{3600} \times 47mA = 0.392mA$

Maximum Power Mode $I_{AVG} = \frac{27}{3600} \times 39mA = 0.293mA$

EcoMode $I_{AVG} = \frac{3600 - 57s}{3600} \times 37mA = 36.4mA$

6M $I_{AVG\,TOTAL} = 0.392mA + 0.293mA + 36.4 = 37.085mA$

Sim 800L :- Total No. of 2A Burst $= \frac{3600}{4.6ms} = 782608$ Bursts.

$782608 \times 577 \times 10^{-6} = \frac{451s}{3600s} \times 2000mA = 250.5mA$

$\frac{3600 - 451}{3600} \times 18.7 = 16.35mA$

Sim800L $I_{AVG} = 250.5mA + 16.35mA = 266.85mA$

Total Current Average $= 266.85 + 37.085 + 30 = 333.935$ mA

Total $I_{Avg} = 333.935mA$

Using 13 Pins on Arduino Nano. 1 Pin draws 20mA

$\therefore 13 \times 20mA = 260mA$

Final $I_{AVG} \approx 333.935 + 260 = 593.935$ mA

$\approx 600mA$

Figure 16, Calculations for average current consumption

So, the picture above shows we will be consuming around 600mA of current. This then takes us to a decisive point where we have to decide how long do we want the system to last for. So, we do further calculation in the following table.

Current Consumptions (mA)	Hours Required	Required Battery Capacity
600	10	6000mAh
600	5	3000mAh
600	3	1800mAh

Table 3, Battery Capacity Requirement

So, the table above shows us the required battery capacity for different hours of runtime. We decided for 5 hours of runtime. Because it seemed a pretty decent number. As this device will be use daily hence it should be charged daily as well. Which makes it pretty fine. We also need to

2.13 TP4056 Charging Module:

TP4056 is a distinguishing component of our project. This module lets us charge the LiPo batteries very conveniently. This module has a built in Integrated Circuit which helps manage all the functionality. Its basic function is to charge the batteries. But important thing to know it how it does that. This module has a USB Mini port for input power so a basic 5V mobile charger can also be used to power this module. But the maximum input voltage rating is 8V[75]. It also comes with two built in LED's, red and blue. It has different charging states. So, when it is charging a red LED lights up, when it has finished charging a Blue LED will light up. As far as the charging speed is concerned, we can potentially charge a fully discharged battery in almost between 1 to 1 and a half hours[76]. The key factor to use this TP4056 module is that it comes with short circuit protection, over charging protection and discharge protection.

2.14 Software

The software will be designed using the Arduino IDE. The code will be written in C language. The code will employ different types of techniques. The objective is to make the code efficient as possible. Along with that we have to make it short which will make its execution much faster. We will be using a combination of libraries to make our objectives a reality. Furthermore, we will make multiple functions which not only simplify the code but make it more efficient. We also have to make sure that we are using error handling and validations so that it keeps making sure that code is running fine.

The main aim of the code will be for it to assist us in obstacle detection. This software will also allow us to take the system into an emergency state in which it will allow us to work with the additional modules and components we are using in the system.

```
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
```

Figure 17, Libraries used for the code

In this software, we will be using two libraries. One is TinyGPS++[77] and the other is softwareserial.[78] They both have two different functions. TinyGPS++ is a library which is designed to parse the information which is provided by the GPS Receiver to the software. If we don't use this library, we will only be able to see NMEA sentences. NMEA is a standard data format for GPS Receivers for data transmission and reception. Therefore, it is not like a language which we can understand. Hence, we use this library to translate that information for us. It is important because we have to extract the latitude and longitude which will be helpful for us in later stages of the software.

Software Serial is another library which we will be using. This emulates the functionality of the hardware serial. This means that we are able to use the digital pins of the Arduino to transfer data and communicate with the modules serially. This also allows us to deploy more modules with a single Arduino. It has different methods which will definitely help us in our project. A few are available(), listen() and end().

---

[75] (NanJing Top Power ASIC Corp, 2024)
[76] (NanJing Top Power ASIC Corp, 2024)
[77] (Arduino, 2024)
[78] (Arduino, 2024)

Available() method checks if a device is available to communicate. This helps us to make a decision and we can then write certain error handling techniques in case a device or the module is not available. Listen() method comes in handy when we want a certain module to act. As multiple modules can't communicate to Arduino simultaneously, we can talk to the modules one by one by calling them using this method. And once we are done, we can just use the end() method to end the communication and communicate with another module.

```
TinyGPSPlus gps;

SoftwareSerial neo6m(4, 5); //Tx, Rx
SoftwareSerial sim800l(7, 6); // Tx, Rx for SIM800L module
```

Figure 18, Declaration of Modules

This image above shows the declaration of the two modules, which tells the Arduino about the location of the modules connected using softwareserial. So with help of softwareserial library it knows where to transmit and where to receive the data. In the first line, we declare gps as an object of the TinyGPS++ library, which we will use later to access different features of the library.

```
unsigned long previousMillis1 = 0;
unsigned long previousMillis2 = 0;
unsigned long lastBlinkTime = 0;
const long interval1 = 500; // Interv
const long interval2 = 700; // Interv
const int ledPin2 = 10; // pin for Le
const int ledPin1 = 11;// pin for Led
const int buzzerPin = 9; // Pin for b
const int trigPin1 = 13; // Ultrasoni
const int echoPin1 = 12; //**
const int trigPin2 = 3;//**
const int echoPin2 = 2;//**
boolean buttonPressed = false; //Butt
int emergency_state=1; // Initializin
int ledState1 = LOW; // Initial state
int ledState2 = LOW; // Initial state
```

Figure 19, Glimpse of Variables Used

Above screenshot shows the globally declared variables. These variables can be used anywhere because they are globally declared which means at the start of the program. Many of them are constant variables as denoted by const, because they shouldn't change during the running of the program which is precaution taken beforehand to prevent any error in the execution stages of the software. We also have the state of emergency set to 1, it can be any number, but for standard we would switch between 1 and 0.

University of South Wales
Prifysgol De Cymru

```
void ultrasonic(int trigPin, int echoPin, int buzzerPin) {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  long duration = pulseIn(echoPin, HIGH); // Reads the echoP
  int distance = duration * 0.034 / 2; // Calculating distan
  Serial.print("Distance from the object = ");
  Serial.print(distance);
  Serial.println(" cm");

  if (trigPin == trigPin1 && echoPin == echoPin1) {
    // If the right side sensor, produce a sound relative tc
    buzzerRight(distance);
  } else if (trigPin == trigPin2 && echoPin == echoPin2) {
    // If the left side sensor, produce a different sound pa
    buzzerLeft(distance);
  }
}
```

Figure 20, Ultrasonic Function

The figure above shows the ultrasonic function we have created. This functions job is to calculate distance by detecting if there is any obstacle. Furthermore, this function also utilizes two other functions, which are used to initiate buzzers depending on the situation. When the distance is certain centimetres, it will turn the buzzer on. This function takes 3 arguments. The reason is we have two different ultra sonic sensor. So, when we want to read value from a specific sensor, we can just put that sensor's specific value in the function. We can write it twice with two different values to read the value from both sensors as demonstrated below.

```
ultrasonic(trigPin1, echoPin1, buzzerPin); // C
ultrasonic(trigPin2, echoPin2, buzzerPin); // C
```

Figure 21, Ultrasonic Function Called

But now this allows us to distinguish between a wider range of area horizontally. So, we have positioned the sensors in a such a way that they cover the centre but also a little bit of right and a little bit of left. Hence, we are able to distinguish between if the obstacle is at a little right or left. Consequently, we have coded following two separate functions.

```
// Buzzer function for the right s
void buzzerRight(int distance) {
  // Define the sound pattern for
  if (distance < 10) {
    tone(buzzerPin, 1000); // Play
    delay(150);
  } else if (distance < 20) {
    tone(buzzerPin, 500); // Play
    delay(150);
  } else {
    noTone(buzzerPin); // Turn off
  }
}
```

Figure 22, Buzzer Funciton

```
void buzzerLeft(int distance) {
  // Define the sound pattern fo
  if (distance < 10) {
    tone(buzzerPin, 1100); // Pl
    delay(500);
  } else if (distance < 20) {
    tone(buzzerPin, 528); // Pla
    delay(500);
  } else {
    noTone(buzzerPin); // Turn o
  }
}
```

Figure 23, Buzzer Left function

These two functions in Figures 22 and 23 allow us to produce sound of different frequencies at different distances which indicate the closeness to an obstacle. But along with this they tell us that in which direction is that obstacle as it will make a fixed sound for object detected in certain direction. Once the person gets used to this, they will easily be able to distinguish between which side the object is and how close it is. The buzzers use tone() and noTone() functions and take the buzzerPin as an argument. Tone function tells the buzzer to make a sound and noTone makes it silent.

It is important to note that the buzzer is at a specific volume because we had to make sure that we didn’t disturb other people around the user as well as the user is effectively able to make use of this technology.

```
void setup(){
  Serial.begin(9600); // Initi
  // Declare the modes for all
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  pinMode(trigPin1, OUTPUT);
  pinMode(echoPin1, INPUT);
  pinMode(trigPin2, OUTPUT);
  pinMode(echoPin2, INPUT);
}
```

Figure 24, Setup Function

This is the setup function in figure 24. In Arduino it is usually called when Arduino powers on. It makes sure everything is setup for Arduino to run the program. So that the communication with Arduino can be initialized using the begin() method. 9600 which is passed into this method is the baud rate which is explained here[79]. Further all other pin modes are declared to distinguish the digital pins. This is necessary because the next function called is the loop function which runs for the whole time repeatedly after this function.

[79] 2.7 Components

```
void gps_location(){
  neo6m.begin(GPSBaud); // Start communication with Arduir
  neo6m.listen(); // Call the Module
  Serial.println("Calling GPS");
  unsigned long startMillis = millis();
  while (millis() - startMillis < 1000) { // Wait for 1 se
    while (neo6m.available()) {
      gps.encode(neo6m.read());
      if (gps.location.isUpdated()) { // Formatting the NM
        Serial.print("Latitude= ");
        Serial.print(gps.location.lat(), 6);
        Serial.print(" Longitude= ");
        Serial.println(gps.location.lng(), 6);
        return;
      }
    }
  neo6m.end(); // End communication with the Module
  }
  Serial.println("No GPS data available, help required");
}
```

Figure 25, Function for retrieval of GPS Location

The function shown above is a function created to retrieve GPS location attributes such as longitude and latitude. It is using the TinyGPS++ library we mentioned earlier. Further it is used in conjunction with the NEO-6M module. Firstly, we initiate the conversation with the Arduino. Then we call the GPS module. The function then waits for 1 second using the millis() function. Millis() function at any point can tell us the time it has taken from when the Arduino has powered on so with checking if 1 second has elapsed we see if the GPS module is available or not using the available() method. If it is it reads the location and then formats it for human reading. After that it ends the communication using the end() method.

```
void sendSMS(String number, String msg) {
  Serial.println("Calling Sim800l");
  sendCommand("AT+CMGS=\"" + String(number) + "\"", ">", 1000, 5)
  sim800l.print(msg);
  sim800l.write(26); // ASCII code for Ctrl+Z to send the message
  buttonPressed = true; // Mark the button as pressed to avoid se
  Serial.println("Button Pressed");
}
```

Figure 26, Function to Send an SMS

This function in Figure 26 is used to sendSMS using the SIM800L module. Its basic function is to send an Attention command to the module using the sendATComm which is another function written to send the Attention commands. It takes in two arguments, one is the phone number, second is the message. We can just store our message in a variable and pass that variable into this argument. But it also changes the buttonPressed flag to true. The reason is that if we press the button once due to hardware debounce it can detect the button multiple times. Hence, we introduced a flag which is changed at a later stage in the emergency state. This function is also dependent on two other functions. Firstly, it is the sendCommand function which includes all the mechanisms to send the command to the module but also includes error handling, which is trying certain number of attempts, if that doesn't happen then send an error message to the system. In case we are unable to send the message, the LEDs would still turn on and blink for visual awareness for people around the user. sendCommand function relies on the checkResponse function which verifies the response which comes from the module. Using two functions helps us to be discrete in functionality and easy for error recognition while programming. We could have implemented another error handling technique in which we can use the Reset Pin in the module and by triggering the reset pin, it would restart the module. But we didn't have enough pins available on the Arduino. The workaround would be simple for that, we can look if any of our other modules could use the analogue pins on the Arduino, if they do then we can definitely implement this. But lack of time and experience we weren't able to execute this technique.

```
void loop(){

  ultrasonic(trigPin1, echoPin1, buzzerPin); // Calculates distance from the 1st Sensor
  ultrasonic(trigPin2, echoPin2, buzzerPin); // Calculates distance from the 2nd Sensor
  delay(100);

  if (digitalRead(BUTTON_PIN) == HIGH && !buttonPressed) {
    emergency_state=0; // in emergency state, state is changed from 1 to 0
    gps_location(); //call gps function for location
    sim800l.begin(9600); // start comm with sim800l
    sim800l.listen(); // call the sim800l module
    String message; // declare local variable for mesasge
    message = "HELP ME!!! https://www.google.com/maps?q=" + String(gps.location.lat(), 6)
    Serial.println(message);
    sendSMS("447888196873", message); // send SMS to this number
    lastBlinkTime = millis(); // Capture the time for counting later for led's

  }
```

Figure 27, Loop Function

The above given screenshots show the loop() function of the software. It runs all the time after the setup() function. We put commands which we repeatedly need to execute. This is divided into 2 parts. This is the first part. We firstly call ultrasonic sensors, both of them as we pass different functions. After that it checks if the button is pressed, by going to the button pin declared initially and checking if it is high. It also cheques if the button is pressed or not. If the button is pressed it goes into the if condition and it changes the emergency state to 0. Then it calls the GPS function to retrieve the location from the GPS module. After this it start the communication with SIM800L module by using the begin() method. Then it calls the module into the function, a variable is declared called message. This variable is a local variable which will store the current location. When assigning the message variable, we format the location which is retrieved using the GPS location function. We also make sure that this variable formats the location in such a way that it is shown as a Google Maps link so the end user can easily locate the person on the maps straight away. After that we call the sendSMS function to send the message to the dedicated number along with the message. Both of them are passed as an argument into the function. Then we capture time for the led's to blink.

```
  // Check emergency state
if (emergency_state != 1) {
  buttonPressed = false;
  if (millis() - lastBlinkTime < 300000) {
    blinkLED(); // Blink LED function to bli
  } else {
    digitalWrite(ledPin1, LOW); // Turn off
    digitalWrite(ledPin2, LOW);
    emergency_state = 1; // Back to normal s
  }

}
```

Figure 28, Emergency State within the Loop

Now we come to the second part of the loop() function. This part basically is the actual emergency state part. It checks if the emergency state has been declared or not. Emergency state was already changed at the start of the function then the value of the emergency state variable was changed to 0. This meant that the system has entered the emergency state. Now once it has, we will ask the system to blink the LEDs for 5 minutes using the blinkLED function. We will use the millis() function to calculate the difference in the time since initiation of the if condition with which we will be able to calculate if 5 minutes have passed or not. If they are passed, then it will switch the LEDs off which is one of the power conservation techniques. The emergency state is also then set to default. But most importantly, this function tells the system that the button is not pressed now. Which means that we now can again press the button. But the emergency state will stay as it is. The reason we wanted to blink these LEDs was to signal visually to the people around us as well that we might be in sort of some help as blinking of LED's is an unusual sight and it is eye catching. Hence, we decided to include this in our project. We didn't use the Buzzer to make sounds when emergency state is declared because buzzer was already being used for Ultrasonic Obstacle Detection.

Chapter 3
Design Phase

Chapter-3: Design Phase

In this chapter, we will be discussing the design aspects of the smart stick. It is very important that we thoroughly scrutinize the way we will design this project. Because one of our aims was to make sure that the device is easy to use for any user. This aim also goes to an extent that the user shouldn't require any complex trainings to be able to use this device as that is the crux of this project that we should find a viable solution for people who are visually impaired.

3.1 Hardware Block Diagram

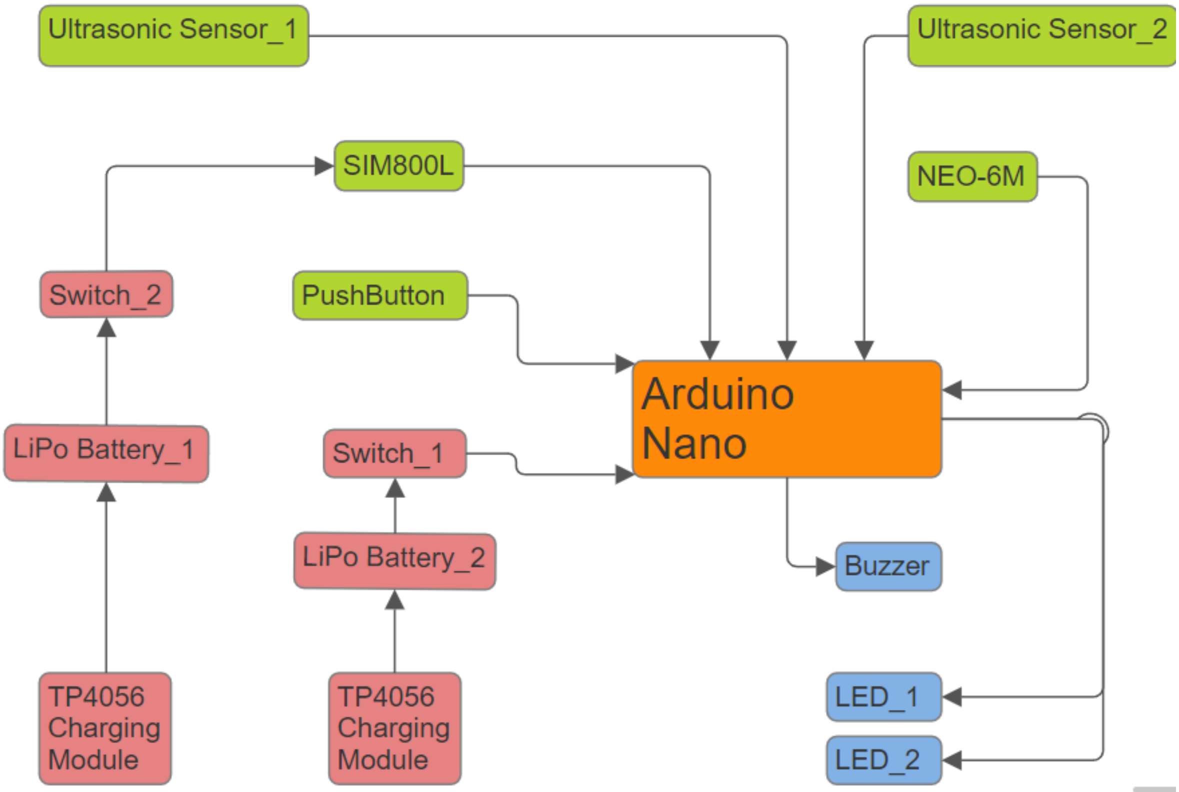


Figure 29,Hardware Block Diagram

The image above displays the block diagram for the hardware. We have 5 devices which will be providing an input coloured in Green to the Arduino Nano. These include 2 Ultrasonic Sensor, SIM800l, Neo-6M and a Push Button. The output depending on the code will be done using Buzzer and two LED's. But the start of a whole process begins from the Switch_1, which is connected between the Arduino and the batteries. This switch is able to power on and off the Arduino. We similarly have another switch, switch_2 which is powers on SIM800L and NEO6M directly from the batteries. The reason behind this is that the SIM800L requires a VCC between 3.4V and 4.4V. Hence, it is not possible to power it from the Arduino. Further we discussed earlier in the section here[80] that SIM800L requires bursts of 2A current. This much current can't be provided using the Arduino. Further there NEO6M doesn't get enough power if we connect it to the 3.3V pin on Arduino Nano hence we connect it to the LiPo battery with it works perfectly fine. Practically, the switches have

[80] 2.9 GSM Module:

been added for protection in case of any mishaps. We have two switches, one to power the module and one to power the Arduino and components connected to subsequently. We have two TP4056 module which has all the required features to protect the batteries from different scenarios like short circuit and overcharging[81]. We practically can keep all the 2 switches on while charging, but this can cause overheating in the batteries hence it is not recommended to charge the device both switches are ON.

Ultrasonic Sensors, SIM800L and NEO6M communicate to the Arduino via digital pins provided in the Arduino and they communicate serially. But this is modified in the code that they don't all communicate all at one because this is not possible in Arduino Nano. Ultrasonic sensors specifically communicate back and forth as the Arduino in a nutshell has to ask them repeatedly "is there any obstacle you found?" But SIM800L and NEO6M are not like this. They operate on their own, and they just take the commands from the Arduino to do certain things. NEO6M, when it is powered on it keeps on tracking the position, but the code is written in such a way that when NEO6M is needed, it sends the current location to Arduino and then it goes back to keep tracking the position. Likewise, SIM800L makes sure that it is always connected to the network, when the Arduino sends a command to it, it performs that action in our case sending an SMS.

3.2 Hardware Schematic

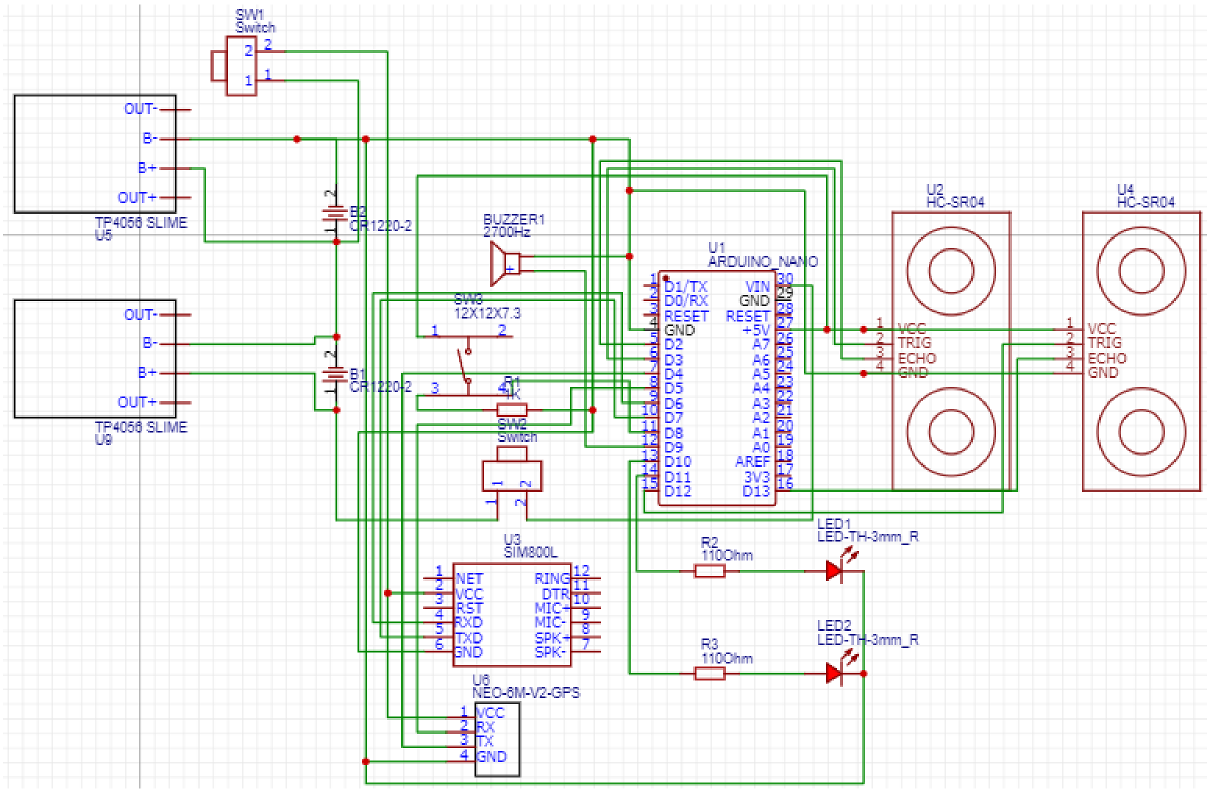


Figure 30, Hardware Schematic

The above is the hardware schematic of our project. It is not much different from the hardware block diagram, but it gives us an insight how it is working electronically. We have a few additional components which were not displayed in the block diagram. These are the mainly the resistors. These

[81] (NanJing Top Power ASIC Corp, 2024)

R2 and R3 are the resistors. These are connected in series with the LED's. These LEDs operate at around 2V. But when the high signal is sent from the digital pin of the Arduino, that signal's voltage is almost 0.7 times[82] the VCC. The calculation of the required resistor are as shown in the image below.

We used Ohm's law to calculate the value of the required resistor which comes upto almost 109ohms. So we can use nearest one would 110 or 120 ohms depending on the availablity but according to series onward E24.

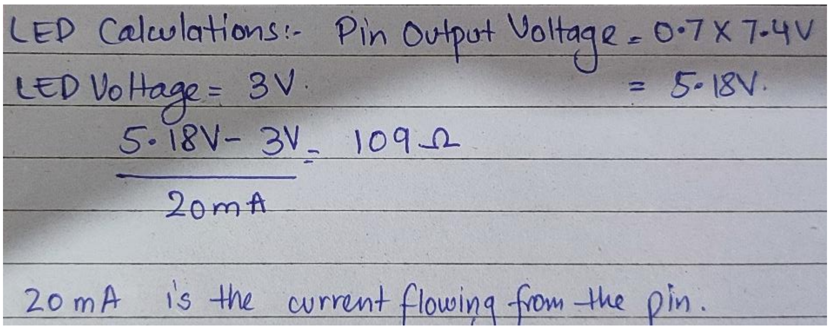


Figure 31, Calculations for Resistor of LED using OHM's Law

Further Resistor R1 is being used in series between the PIN 3 of the button and the Ground. This makes it a Pull Down resistor. The reason to add this resistor in this configuration here is that we have to make sure that the logic level is always low. When the switch is not pressed the current is flowing from the ground to digital pin. As ground is at assumed at 0V this presents us a low logic. But as soon as the switch is pressed the current flows from the VCC to the digital pin indicating that the signal is high hence we are able to efficiently differentiate between a low and high signal.

In this schematic we had to make sure that we were using a common ground pin. The reason to use a common ground pin is that every module has to compare their value to a certain logic level. Hence it all needs to be common. Let's say if I connect to a ground pin which is not connected to the Arduino, my module will not work properly because we receive the signal from the Arduino, which is high, hence we need to complete the circuit using the ground so we can have a reference value otherwise we will not be able to distinguish between signals.

We have connected two batteries in series which makes the total voltage to 7.4V. The reason to take such a step was because we needed to operate Arduino. But using SIM800L meant that we need another power source. So rather than using two different power sources we decided to use two different batteries. We can power Arduino using both of them in series, and we can also power SIM800L using a connection from one of the batteries. But we had to do this carefully because if we connect the SIM800L to the battery 1, the first in sequence, the SIM800L didn't work properly because there was no common ground. Hence, we have to connect it to the second battery as the ground of that battery was connected to the Arduino Ground.

As we are using two different batteries, we are using two different TP4056's to charge them. The reason is pretty simple. Safety. A single TP4056 won't charge both of the batteries. If we use two of them, we will be able to protect the user from many hazards and make the system more efficient. There will be no chance of the batteries discharging into each other as when the circuit off as we have

[82] (Microchip, 2015)

a switch between Arduino VIN and the battery's VCC, batteries won't be able to discharge into each other. Further this way we will be able to maintain the battery life of each of the battery because we will be charging them separately which results in equal charging. Using one module if that is possible, would have a probability of not being able to charge the batteries equally which can damage the battery life.

Last but not least, Figure 30 shows us that we are using two different switches in this project. The reason behind this is related to the batteries again. As we use two different batteries to manage the voltage and current for the modules as well as the Arduino at the same time hence if we didn't use two switches, the circuit will keep running. One switch is the one which powers the Arduino as well as the Buzzer, LEDs, and Ultrasonic Sensors as they are connected to the Arduino for power. The other switch is between the second battery and the SIM800L module. We also gave a connection to NEO6M via the VCC of SIM800L which provides the required voltage to NEO6M as it was not operating using the 3.3V provided by the Arduino.

3.3 Component List and Bill of Materials

Item	Supplier	Part Number	Description	Quantity	Price inc. VAT	Total
1	Farnell	1848691	Arduino Nano	1	19.2	19.2
2	Kunkune	Sim800L	Sim800L	1	4.9	4.9
3	Kunkune	Neo-6M	Neo-6M	1	4.9	4.9
4	Farnell	4162009	Ultrasonic Sensor HC-SR04	2	3.11	6.22
5	Amazon	KY-012	Buzzer	1	1.05	1.05
6	Amazon	TP4056	TP4056 Charging Module	2	0.8	1.6
7	Amazon	LP503870	LiPo Battery 3.7V 1500mAh	2	6.12	12.24
8	Amazon		Stick	1	11.36	11.36
9	Amazon		Enclosure	1	7	7
10	Amazon	KCD1	Button Round 2A	2	1.2	2.4
					Total Cost:	70.87

Table 4, Bill of Materials

The table above shows the main components used in this project. The total of cost of these components is £70.87. One of our objectives was to make this Economical. In our opinion, this is a very big achievement. Although this doesn't include labour costs and small components like button, LEDs and resistors. Costs for small components are negligible for this stage as they are bought in bulk. But even if they were to be bought in larger quantities, the price per component would be negligible but then it would be much better to add that in the production cost due to the scale of purchasing.

Secondly, this sets a base to work from for commercial manufacturing. Our Supplier are retailers and not manufacturers. But when it comes to commercial manufacturing, you goal is to make the thing as cheap as possible. Hence you order the components directly from the manufacturer. We can also ask the manufacturers to modify the modules if we have to mass assemble the product. The reason is that this will also help to cut costs. For example, we don't need the whole Arduino Nano board. We can just use the same microprocessor and flash it with bootloader and firmware. Furthermore, we can customize the hardware to save costs. The same can be done with other components.

The reason we had to buy from Amazon and eBay was they were much quicker in delivering the components. And we are just building one smart stick so for this case, it is justified. But it is not feasible at all for 100 smart sticks. We still believe that if we had the time, we would be able to cut costs, by choosing other suppliers.

3.4 Flowchart

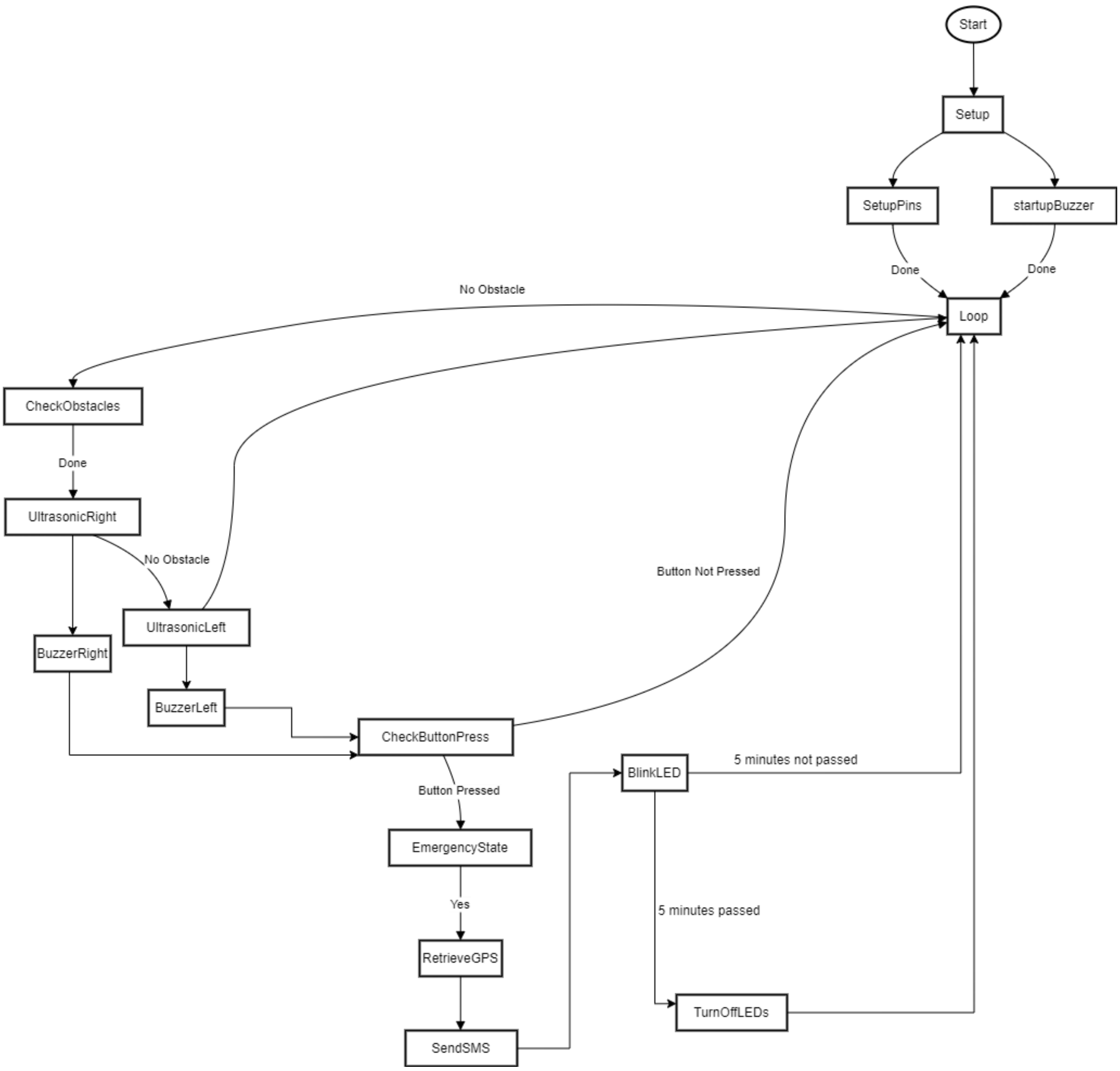


Figure 32, Flowchart of the software

The Figure 32 shows the flowchart of the software, how it works. So, the first thing it does is ask the ultrasonic sensors to scan for obstacles. We basically have two Ultrasonic sensors so one which covers a bit of right and the other which covers a bit of left. Therefore, if there is an obstacle found, it will turn the buzzer on, but it will use the appropriate function to generate a distinct sound for the conditions we have explained above here[83]. If no obstacle is found on either side, it will again start scanning it.

But it also checks that if the button is pressed or not. If the button is pressed, the emergency state changes to 0 from 1. The software then calls the NEO-6M serially and gets the location. The location is being saved in a variable. So, whenever the system needs the current location, it will call the module to get the current location.

Next, it will end its communication with NEO-6M and call SIM800L module. It will then format the location retrieved from the GPS into human readable form. The formatted message will be put into a String Variable. This variable will be forwarded to the SIM800L using AT Command and the message will be sent to the dedicated number.

After it is done, we have another condition which check if the state is not equal to1, if it is, it exists the condition, but if it is not, it checks if the time is less than 10 minutes or not using the help millis() function and another variable. If it is less than 5 minutes, it blinks the LEDs, while keep scanning for the objects. If the time is over 5 minutes, LED's stop blinking, and the software returns to its default state.

After it is done, we have another condition which check if the state is not equal to1, if it is, it exists the condition, but if it is not, it checks if the time is less than 10 minutes or not using the help millis() function and another variable. If it is less than 5 minutes, it blinks the LEDs, while keep scanning for the objects. If the time is over 5 minutes, LED's stop blinking, and the software returns to its default state.

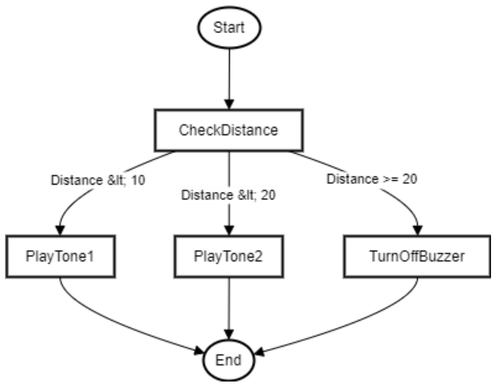


Figure 33, Buzzer Flowcharts Summarized

Figure 33 shows how the buzzer works. We actually have two function buzzerRight and buzzerLeft designated for the ultrasonic sensors. So, they compare the stored value with the new value in the distance variable. If is in the range which is specified here, then they play certain frequency for each range.

[83] Figure 22, Buzzer Funciton

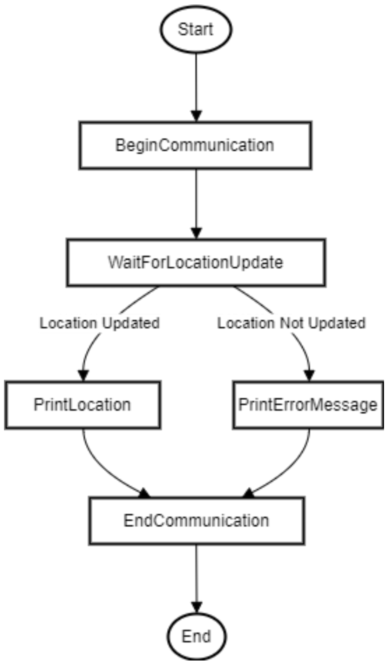


Figure 34, GPS_LOCATION function flowchart

Figure 34 gives us an insight into the retrieval of GPS location. We have separate function to retrieve the location of the GPS which is later called while in the loop function. We firstly start the communication with NEO-6M and then wait for the location to update. This only happens if the module is available, if the location is not updated it means, the satellite connection is not established hence the module is not available to give any information on the location. If the satellite connection is established, blinking Red Led on the module indicate this, then the location is considered updated which is then printed into the serial monitor as well as stored in the message variable which is to be forwarded to the the sendSMS function later as an argument.

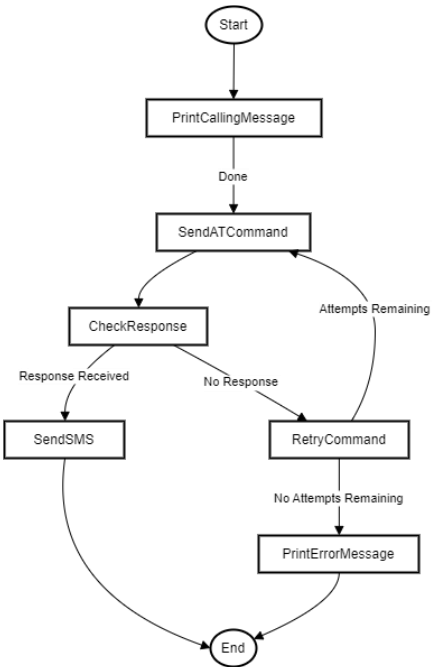


Figure 35, sendSMS function flowchart

The figure above displays the working of the sendSMS function. Before calling this function, the communication is already established in the loop function between sim800l and Arduino. When the function is called it sends an attention command to the module which asks the sim800l to send a message to the number which was passed in the argument and the message should be what is in the variable which was passed in the argument as well. It then check's if the module responds correctly, if it does not, let's say gives an error it will retry command while making certain number of attempts, if the message still isn't sent, that means there is some issue with the network connection or the simcard. This function relies on two other functions which are sendCommand and checkResponse functions whose function is self-explanatory.

3.5 Testing Strategy

This is a very crucial part of the project. This ensures that we make sure that the product works properly and exactly how we want it to work. So, for that reason, we have to create a testing strategy to make sure that our objectives are achieved. In a testing strategy, we take the product towards different testing phases which are relevant to the device and its components. So, we can push it towards its limits. So, there are few things which we will need to test, and which are very important to make sure that this device works perfectly fine.

We tested Navigational Accuracy of the system. It is really crucial because if the system doesn't provide an accurate location it can lead to a lot of panic in state of emergency. We went to different areas to check how accurate the location is. Testing it in different areas also enabled us to test GSM module. We had to make sure it is able to connect to network at different places.

We also need to make sure it is able to detect obstacles. What we can do is that we can firstly, detect obstacles statically, different obstacles. Then we can test the thing whilst walking at different places to see how accurate is. We will also see if the buzzer works fine. We will focus on positioning the buzzer at an accurate place where the user is able to hear it. We also measured the distance between the ultrasonic sensor and the obstacle to make sure that the distance between the two is what is programmed into the Arduino. Furthermore, buttons were tested by switching on and off that work fine. Along with that charging ports were tested by testing the product to full charge.

Chapter 4

Implementation and Testing Phase

University of South Wales Prifysgol De Cymru

Chapter-4: Implementation and Testing Phase

4.1 Testing Phase

We had everything to go ahead with. The testing was required at both the hardware side as well as the software side. Our approach was to first test each of the module individually, hardware wise and then with the software. In hardware testing, we firstly did visual inspection to make sure that all the components on the modules are soldered properly. Then we connected them to Arduino one by one and tried to code them. We have different versions of code, some of them worked and some of them didn't not. But we have saved these versions in the documentation. Whilst we tested the components individually, we used a multimeter to make sure that all the readings we are getting are correct. We had to make sure that we are giving the right amount of voltage, and the wires are connected properly. In addition to this we had to make sure that there is no short circuit. We tested the components on a breadboard. After all the testing was done, we decided to combine all the components together. Here a common issue we faced was that Arduino was getting shut down automatically. The reason behind this was that the common ground. We weren't doing it properly; the Arduino would then shut down. But we then had to make sure, the grounds are properly connected to each other. After everything was connected and working together, we had to take the readings again to make sure every component was given the required power. After all of this was done, we had to replicate it to the schematic.

The schematic was replicated onto a Veroboard rather than a PCB. The reason was that the PCB were ordered but they didn't arrive on time. Hence, it was our instinct that came into play, and we didn't wait for the PCB. But the reason we were instinctive was that this is one of those things we knew, if it didn't go well, we were prepared for that although it is not a direct risk, yes in risk assessment we did assume the late delivery of the components, but this is something we were expecting when we ordered the board. Working with the Veroboard was very complicated, as it was our first time. The wires are all over the place, so we had to be very careful. Also, we had to make sure that the wires were definitely soldered properly, a single connection could short the whole circuit as well as make the project malfunctioned. Furthermore, an approach we took to separate the power section from the component section was to place the relevant components at a distance rather than very close to each other. This made sure that there is no connection between the power components and other components directly except power rails. Power rails were one of the parts of the board which we were able to use very efficiently to manage the power.

For the software, testing was done initially one by one. For each module, we had to write some code initially, we had to look at the internet for different techniques in case if the code didn't work. We firstly made a code file for each module. Which was made after a number of revisions. This also helped us later on because if the project wasn't working properly, to debug it we can just run that specific code and check if the component is working properly. The code was then combined in a single file and run to check together. Furthermore, we performed version control by saving different versions of the file to make sure that we have the previous version as well along with the final one. This allowed us to correct our mistakes as well as keep record of the changes in case we find an error later on. The version control also lets us keep a backup as well in case of any files which get corrupt, or the code messes up.

4.2 Implementation Phase

In the implementation phase, the goal was to assemble all the components. The components were soldered on the Veroboard using a Soldering Iron and solder. Two types of wires were used to connect. The LEDs were connected to Arduino via a thinner wire compared to the wires used for rest of the connections which were 24AWG. This was enough to handle 2A current which was the maximum we would be working with. We also used a silicone glue gun to secure the connections in the place. The buttons and charging modules were also placed and fixed using the silicone glue gun. Silicone works good with plastic, so it was a adhesive to use. We also used an enclosure to put all the things into. It kept the project neat and clean. To make it further neat, we used a wiring sleeve to make the wires look tidier which go down towards the ultrasonic sensor. We also used glue gun to glue the ultrasonic sensor in place. The enclosure is tied to the stick using zip ties and zip ties were glued as well so they don't move.

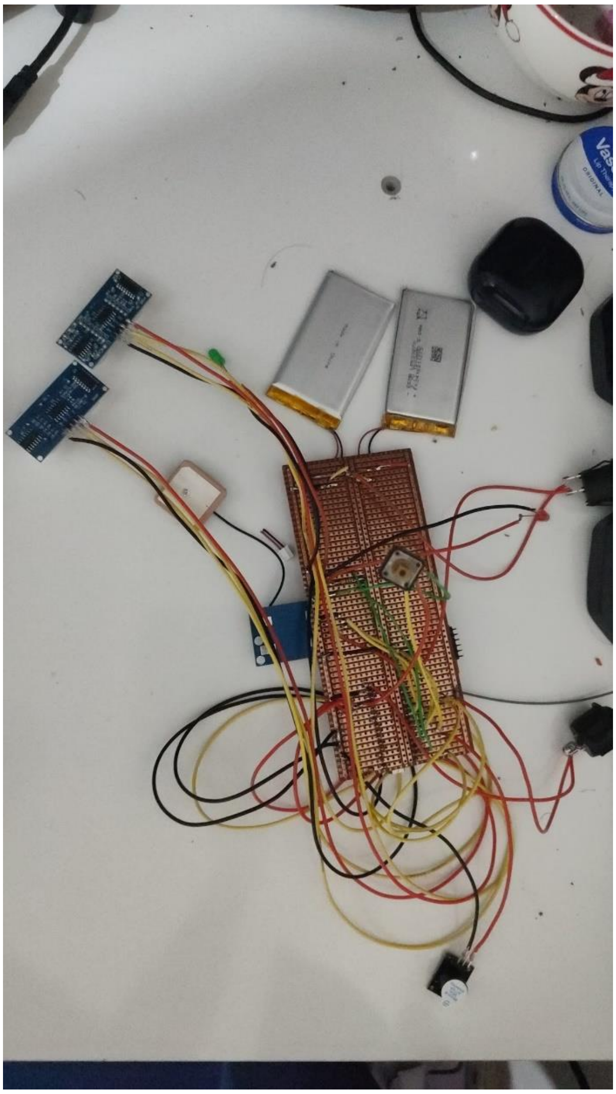


Figure 36, Before assembly

The image in Figure 36 shows the condition of the project which was right after everything was soldered. This gave us an opportunity to completely test the system entirely. We also made sure all the solder joints were completely soldered.

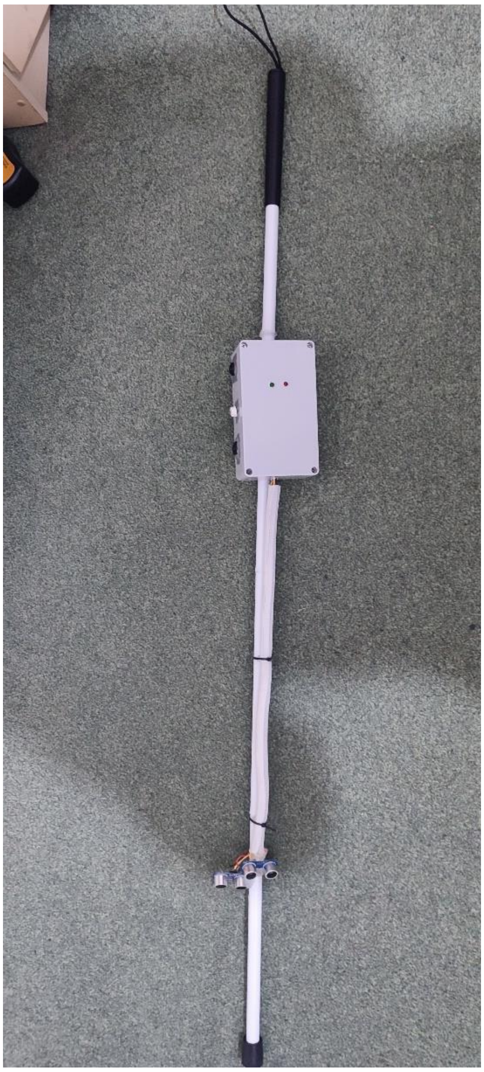


Figure 37, Actual Stick

Figure 37 displays the actual stick after assembly. It can be clearly seen that the project was made much tidier than it looked before. We added an enclosure which was tied to the stick. This enclosure held the PCB and other components. Further a sleeve to hold the wires was added which go down to the ultrasonic sensors. It can clearly be seen that the sleeve is camouflaged and doesn't ruin aesthetics. After we had the Ultrasonic sensors at the bottom, they weren't attached lower than that because of the possibility of them getting dirtier.

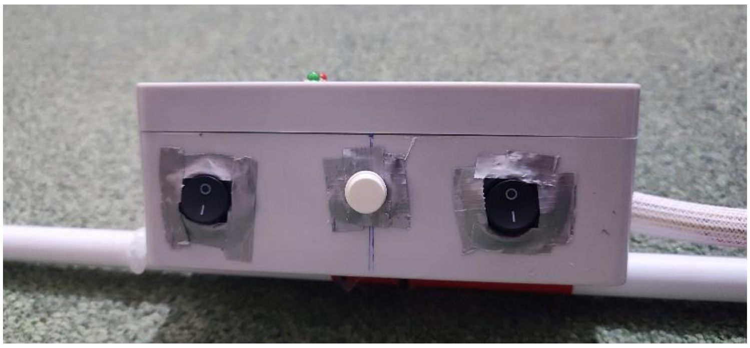


Figure 38, Side Buttons

In Figure 38, the buttons were carefully placed on the side. The middle button works to send the location as SMS to the designated mobile phone number. The enclosure is sealed shut with 4 screws provided with the enclosure.

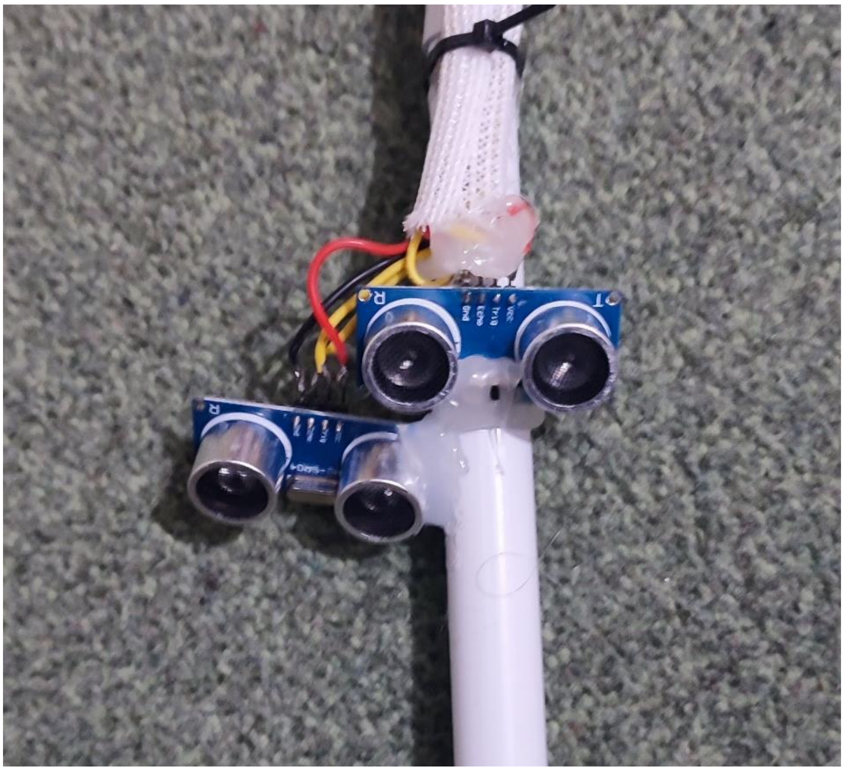


Figure 39, Sensors placement

In the image above, it is shown that the sensors are place very precisely at certain angles. This was a result of test and try approach. This configuration gave us the best results. We were able to have some part of both side and the centre scanned using this configuration.

Chapter 5
Conclusion

Chapter-5: Conclusion

To conclude everything, I think we managed to make a successful working Smart Stick for visually impaired people. This project was of immense importance due to the severity of lack of proper practical solutions for the problem. Hence, we think we were able to succeed in this project very convincingly. This project will definitely help the blind people enhance their mobility as well as their freedom of movement. This will also contribute to the mental and physical health of the visually impaired people as they will be able to move freely without hesitation as they know that they have can press the emergency button in case of any emergencies which provides a sense of security. This project also aims to provide awareness to other people who are nearby the user that there is differently abled person around them which will definitely let other people accommodate the user as the buzzer which is integrated in this project will not only tell the user that there is an obstacle but along with that it will also aware other people around the user. Furthermore, the blinking LED's will enable people around the user to see that the person is in emergency as it will happen only when the emergency button is pressed. Hence this will implicate that the user is an emergency.

Despite being this a very straightforward project there were a lot of hiccups throughout the project. The most difficult thing about this project was that the lack of variety of components. This meant that we had very limited components to choose from. Furthermore, we had to make the thing as cheap as possible while keeping the system reliable. The most prominent example is SIM800L module. This module supports 2G only. This thing is not explicitly mentioned anywhere which did lead us astray for a while and this wasted very important time of ours as the module didn't work. Many of the network providers in the UK have stopped their 2G services[84]. But some do provide 2G, but there is no operator which provide GPRS or 2G Data services. We did our research before choosing this module, but this is an extensive and a very external issue we faced. We also had additional plans with this module which will be discussed later in the last section. But we had to then scrap this plan and change it in the middle of the project. It was a little disheartening, although we still found a solution for this.

Initially using a buzzer was not in our plans. We mentioned in our interim report that we will be using a module called DFPlayerMini. That was a MP3 module which plays MP3. We planned that we will use 3 ultrasonic sensor for different directions and then it will play a relevant file relative to the direction of the object. But then there was an issue. When the module was connected with the Arduino, it wouldn't play the file. We had ordered two of them. We tried the other one but it didn't work. Then we tried the code using a different Arduino library. But that didn't work either. We placed debugging prompts in each version of code to see if the code is working fine. The debugging prompts were telling us that the Arduino is unable to establish a serial connection between the Arduino and the DFPlayerMini. We used serial.available()[85] command of Arduino to check and a few further debugging code lines. But it still didn't work. We then tried a different Arduino, an Arduino Uno, we followed the same procedure as above and the DFPlayerMini was still not able to play sound when it was asked to by the Arduino. Along the way, we also made sure that the RX and TX pins are connected perfectly. In addition to this we were monitoring the Voltage across the input pin and transmissions pins, but they were perfectly fine with both of the modules. We then decided to give one more try and order a different but same module. It's not made by DFRobot but it works the same way using a different library. It's known as MP3-TF-16P. It uses a slightly different chip apparently, but it

[84] (Ofcom, 2024)
[85] (Arduino, 2019)

works the same way. But it still didn't work despite all the connection were perfect and we monitored the voltage across the pins as well.

Yes, there was another possibility that the SD Card which was inserted in the module was not properly formatted or it was corrupt. Along with this, we were using earphones so the earphones might have been faulty. The last thing we could do was to use the second ground pin provided in the module, we tried the second pin but it didn't work either. When we were wiring one of these modules accidentally a wire touched the I/O pin which is a separate pin on this module. This triggered the module, and it started playing the audio which was inside the SD Card. The indication was given by the module as it's blue light turned on. Furthermore, the audio was being played through the earphones. Resultantly, it proved one thing that the module works fine in general. Consequently, there was definitely some issues between the communication infrastructure. It might be the code although the code was thoroughly checked over and over again and put through the compiler and seemed fine. So, we had to scrap this idea and look for other alternatives.

The alternatives which came into mind were using a separate MP3 only module, which we can connect to a speaker. We decided not to go down this way because we were extremely late due to the unexpected behaviour of these modules. So, a buzzer was a quite a good option. Other than that, we were almost on time and in line with our timeline. The only problem was the delay due to very unexpected errors.

5.1 Outcomes

The objectives were listed in this section[86] earlier. We were able to successfully achieve all of our objectives.

1. Reliability and Safety

We were able to ensure that the system is Reliable due to the software and measures we have taken up in case of emergency.

Switches helped us make sure there are no excessive supply of current and the power is in control.

TP4056 helped us regulate the charging process so that the batteries are safely charged. It has safety features like short circuit protection, overcharging protection and discharging protection. This makes it safer.

Button Press Debounce was handled by add a flag which protects the button sending multiple positive signals to the microprocessor. Further this would allow the user to send the SMS multiple times. This makes system more reliable.

LED would turn on and blink in emergency state for visual awareness around the user this makes it reliable for public use as well.

2. Economic Feasibility

In this section here[87], we have demonstrated that this project is economically viable hence we have completed another objective. The products discussed earlier in the competitiveness section are much more expensive, at least 3 times more expensive.

[86] 1.3 Objectives

[87] 3.3 Component List and Bill of Materials

3. Obstacle Detection

Obstacle Detection was made possible with the help of HC-SR04 sensors which were paired with a buzzer to let the user know that there is an obstacle. We were also able to distinguish between the object location using different frequencies which passed on to the buzzer.

4. Efficient Communication

This was made realistic by using the GSM SIM800L module. If it was not used, we wouldn't be able to communicate with the designated contacts in case of emergency.

5. Location Tracking

We achieved e location tracking, using the GPS NEO-6M module. Whenever we need to retrieve our current location, we would do it with the help of this module.

6. Power Management

Power Management was crucial for such a project as we wanted to have a long battery time. Furthermore, addition of TP4056 module made it possible to help in prevention of any incidents. It also facilitates charging of the batteries in an efficient manner.

5.2 Reflection

Working on a project like this has been an enlightening journey for us. It gives us an insight and makes us feel about the problem other human beings might face. It also projects a sense of responsibility on us to contribute to the society. Along with this, while doing this project we were able to go over our abilities and skills. We learnt many new skills. We made mistakes and also learnt that how to counter those mistakes in a patient and sensible manner. We were able to make ourselves more patient with the things. In this project, risk assessment and awareness of the sensitivity of these risks was really important. We did order our PCB, but it didn't come on time. We had to be a little instinctive about this hence we build the circuit on Veroboard. If we wouldn't do this, we wouldn't be able to complete the project on time. This also made us to actually care about the deadlines and use these deadlines to be productive rather than getting lazy and doing things at the very end. Also working on Veroboard was an interesting experience in itself.

We did order spare modules and components when we ordered the parts. But apparently the spares weren't enough. As told earlier DFPlayerMini didn't work at all. But sim800L, the 2nd one did work. We ordered 1 in the first batch which malfunctioned but then we ordered another batch which included 2 of these modules. One of these was physically defective but the remaining one worked.

Working on the project all alone lets you know how much work actually goes on into the products which are commercially available. Reading research papers and researching products lets you know which things are important for products to be in the market commercially. Like Patents, Manufacturing, Certifications etc. Having a single member team, also limits the thinking capacity for a project. You are the only one who knows what you are actually doing, and you know the gravity of the situation if you are in a trouble. Lack of creative ideas consumes time to solve any problem which occurs.

Researching was a crucial part of this project. It takes a lot of time. Anybody would have to consider many aspects while researching. We could obviously do better in this department as well, and now we do know how to do it in a better fashion compared to this project before, that is something we learned.

Doing research thoroughly will mitigate many mistakes which would come up later on. Hence, we do think it is important to proper research regarding the components and design techniques.

5.3 Suggestion for further work

I think this project can be used a beacon of support in the field of mobility assistance technology. We can use this project as a base to progress further as there is a lot of space of improvement available. There are some ideas of ours which we were not able to execute due to time constraints and lack of experience in certain fields. We talked about that we had unfinished plans with SIM800L here[88]. We planned that we would create an app for smartphone. This app will be used to track the live location of the user. This app will primarily be used by a guardian or a family member. The way wanted to do it was using the abilities of the SIM800L of 2G Data. Using this ability, we will send the data to a server and use the app to retrieve the data from the server. We would have displayed the data using Graphic User Interface to make the app more user friendly. Furthermore, this app can be used as a base to provide many more features. We can firmly integrate call features from the app to the SIM800L, so the user can be contacted by their guardian who has the app installed as SIM800L.

For sake of improvement, we can use the new SIM79600CE[89] module which support 4G communication as these are the modern day standard for communication. Adding these components will enable us to think much more and allow us to integrate newer technologies.

We would definitely work to make the audio communication in this system much better in the future. Along with what we intended to do with the DFPlayerMini, telling the direction of the object, we would like to add the distance from the object which should be told to the user. In addition to this, we can also add voice communication to give commands to the smart stick which would enable the user to perform multiple actions like asking for directions so the audio then could be played through a Text to Speech module. But this would have to research over the option we can integrate efficiently in this system.

The last and foremost thing which will distinguish this project is if we add obstacle detection literally. We want the system to tell the user that there is a pole in front of you, or there is a bench at 11'o clock. This will take obstacle detection at a competing level with other technologies. As all other notable technologies are integrating Artificial Intelligence to improve the systems. There will be a lot of extensive work and research required to make this idea a reality. But for now, we can just express how the procedure may look like. We might need to add a camera instead of a sensor. We know that facial recognition is a established technology now and AI is being used to make this process efficient. We just need to provide Artificial Intelligence with a database of maximum things possible which are placed on roads or streets. Once the model is trained with those, we can implement it in the system. But there might be a need of change of many components even the microcontroller because of the high processing needs of this technology. Further, we can integrate the camera with text to speech module as well. For example, if the camera sees a sign which says, "Downing Street", the system can then tell the user that "the downing street is on your left". Integrating Artificial Intelligence with this technology will open doors for many new opportunities. As we talked about 3G and 4G communication modules, we can integrate them with the artificial intelligence which will enable to send and receive data from a common database. This will expose the system to self-induced learning, which means if the AI sees an obstacle, which it never had seen before, it can identify it and store that into a database. When another user's system sees a similar kind of object it can tell the user that it is

[89] (DFRobot, 2024)

that thing. This will reduce the burden on the system when integrating the database of things as we can't have all the things in the world, but we should aim for the maximum.

Using a DFPlayerMini is definitely a perfect idea to avoid using a buzzer as buzzer was never our preferred option. But we still used the creativity in our code to make sure we are distinguishing among the obstacles at certain directions and distances. Also, that would allow us to use a earphone which is more private than a buzzer as the buzzer maybe annoying to the people around us. Although we made sure that the volume of the buzzer is such that the people around the user don't get disturbed and the user is able to know what the buzzer is telling.

An innovative idea to add into the power section of this product would be to also add a Solar Panel charging option. The batteries can be charged by solar energy. This can be beneficial and reduce the burden of charging the device over and over again by wire. This aligns with the current industry trend but for this we would need an appropriate panel with the required output power.

To give a more interactive experience to the visually impaired users, we would love to add Brail pattern embossing on the body of the product to indicate which switch is which.

Considering the design of the circuit, we could definitely work on it to make it much better and less complex but only if we had enough time. We can integrate step down transformers, whilst using a single 12V rechargeable LiPo battery in which we would be able to power the sim800l, neo6m and Arduino simultaneously without additional need of another charging module.

References

Aircom International, 2002. *https://sites.pitt.edu/~dtipper/2700/2700_Slides5K.pdf.* [Online]
Available at: https://www.pce-fet.com/common/library/books/50/7377_1-GSMOverviewAircom.pdf
[Accessed 15 March 2024].

Arduino, 2019. *Serial.available().* [Online]
Available at: https://www.arduino.cc/reference/en/language/functions/communication/serial/available/
[Accessed 2024 March 2024].

Arduino, 2024. *Arduino Nano 33 IoT.* [Online]
Available at: https://store.arduino.cc/products/arduino-nano-33-iot
[Accessed 17 March 2024].

Arduino, 2024. *Arduino Nano.* [Online]
Available at: https://store.arduino.cc/products/arduino-nano
[Accessed 1 03 2024].

Arduino, 2024. *Arduino Nano 33 BLE.* [Online]
Available at: https://store.arduino.cc/products/arduino-nano-33-ble
[Accessed 17 March 2024].

Arduino, 2024. *Arduino Nano 33 BLE Sense.* [Online]
Available at: https://store-usa.arduino.cc/products/arduino-nano-33-ble-sense
[Accessed 17 March 2024].

Arduino, 2024. *Powering Alternatives for Arduino Boards.* [Online]
Available at: https://docs.arduino.cc/learn/electronics/power-pins/
[Accessed 17 March 2024].

Arduino, 2024. *Product Reference Manual.* [Online]
Available at: https://docs.arduino.cc/resources/datasheets/A000005-datasheet.pdf
[Accessed 17 March 2024].

Arduino, 2024. *Serial.begin().* [Online]
Available at: https://www.arduino.cc/reference/en/language/functions/communication/serial/begin/
[Accessed 18 March 2024].

Arduino, 2024. *Softwareserial.* [Online]
Available at: https://docs.arduino.cc/learn/built-in-libraries/software-serial/
[Accessed 22 March 2024].

Arduino, 2024. *TinyGPS.* [Online]
Available at: https://www.arduino.cc/reference/en/libraries/tinygps/
[Accessed 22 03 2024].

Arduino, n.d. *Arduino Nano.* [Online]
Available at: https://store.arduino.cc/products/arduino-nano
[Accessed 02 November 2023].

Arduino, n.d. *Arduino Nano Data Sheet.* [Online]
Available at: https://docs.arduino.cc/resources/datasheets/A000005-datasheet.pdf
[Accessed 04 November 2023].

Arduino, n.d. *Libraries.* [Online]
Available at: https://www.arduino.cc/reference/en/libraries/
[Accessed 02 November 2023].

Bhatta, N. M. P., 2017. RADAR and its applications. *International Journal of Circuit Theory and Applications,* Volume 10, pp. 1-9.

Components 101, 2021. *SIM800L GSM Module.* [Online]
Available at: https://components101.com/wireless/sim800l-gsm-module-pinout-datasheet-equivalent-circuit-specs
[Accessed 19 March 2024].

Cytron Technologies Sdn. Bhd, n.d. *Data Sheet for Ultra Sonic sensor.* [Online]
Available at:
https://www.researchgate.net/publication/258883212_Using_Ultrasonic_and_Infrared_Sensors_for_Distance_Measurement
[Accessed 3 November 2023].

Cytron Technologies, 2013. *User's Manual.* [Online]
Available at: https://grobotronics.com/images/companies/1/HC-SR04Users_Manual.pdf
[Accessed 18 March 2024].

DFRobot, 2024. *SKU:TEL0124.* [Online]
Available at: https://wiki.dfrobot.com/SIM7600CE-T_4G(LTE)_Shield_V1.0_SKU_TEL0124
[Accessed 21 March 2024].

Discant, A. a. R. A. a. R. C. a. B. A., 2007. *Sensors for Obstacle Detection - A Survey.* s.l., IEEE, pp. 100-105.

Electronic Clinic, n.d. *Time-of-Flight ranging Sensor.* [Online]
Available at: https://www.electroniclinic.com/wp-content/uploads/2021/06/TOF10120-Datasheet.pdf
[Accessed 11 February 2024].

Farnell, 2024. *BANNER ENGINEERING QT50R-US-AFH.* [Online]
Available at: https://uk.farnell.com/banner-engineering/qt50r-us-afh/radar-sensor-24m-time-of-flight/dp/3872615
[Accessed 10 March 2024].

Farnell, 2024. *DFROBOT SEN0259.* [Online]
Available at: https://nl.farnell.com/en-NL/dfrobot/sen0259/lidar-laser-range-sensor-tfmini/dp/3769959
[Accessed 10 March 2024].

Farnell, 2024. *DFROBOT SEN0395.* [Online]
Available at: https://uk.farnell.com/dfrobot/sen0395/mmwave-radar-board-arduino-board/dp/3879712?cjevent=0c781db2c8ce11ee810600c00a18b8f9&cjdata=MXxZfDB8WXww&CMP=AFC-CJ-UK-7947610&gross_price=true&source=CJ
[Accessed 11 February 2024].

Farnell, 2024. *MULTICOMP PRO PSG04176.* [Online]
Available at: https://uk.farnell.com/multicomp-pro/psg04176/ultrasonic-distance-sensor/dp/4162010?gad_source=1&gclid=Cj0KCQjwqpSwBhClARIsADlZ_TmB_w2hFE6JMCdxHqL1Vu6zICbWVoY1NM8OeR4zmuVDmVa5nIy8-TUaAhuJEALw_wcB&mckv=_dc|pcrid||plid||kword||match||slid||product|4162010|p
[Accessed 10 March 2024].

Federal Aviation Administration, 2022. *Satellite Navigation - GPS - How It Works.* [Online]
Available at:
https://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/navservices/gnss/gps/howitworks#:~:text=To%20accomplish%20this%2C%20each%20of,that%20provide%20extremely%20accurate%20time.
[Accessed 14 March 2024].

Gbenga, D. S. A. a. A. A., 2017. Smart walking stick for visually impaired people using ultrasonic sensors and Arduino. *International journal of engineering and technology,* 9(5), pp. 3435-3447.

GeeksforGeeks, 2020. *Difference between FDMA and TDMA.* [Online]
Available at: https://www.geeksforgeeks.org/difference-between-fdma-and-tdma/
[Accessed 15 March 2024].

GISGeography, 2024. *How GPS Receivers Work – Trilateration vs Triangulation.* [Online]
Available at: https://gisgeography.com/trilateration-triangulation-gps/
[Accessed 14 March 2024].

GPS.GOV, 2022. *Timing.* [Online]
Available at:
https://www.gps.gov/applications/timing/#:~:text=Each%20GPS%20satellite%20contains%20multiple,receiver%20to%20the%20atomic%20clocks.
[Accessed 14 March 2024].

Gupta, M. P. A. a. A. R., 2018. *Smart Stick for the Blind and Visually Impaired People.* Coimbatore, s.n., pp. 542-545.

L. Koval, J. V. P. B., 2016. Distance Measuring by Ultrasonic Sensor. *International Federation of Automatic Control,* 49(25), pp. 153-158.

Massachusetts institute of technology, 2020. *Batteries & Energy Storage.* [Online]
Available at: https://ocw.mit.edu/courses/2-60j-fundamentals-of-advanced-energy-conversion-spring-2020/ed4b85c28d16b5cc2dd0d2635a5723f4_MIT2_60s20_lec11.pdf
[Accessed 9 March 2024].

Microchip, 2015. *8-bit AVR Microcontroller with 32K Bytes In-System.* [Online]
Available at: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf
[Accessed 13 March 2024].

Mohammad, T., n.d. *Using Ultrasonic and Infrared Sensors for.* [Online]
Available at:
https://www.researchgate.net/publication/258883212_Using_Ultrasonic_and_Infrared_Sensors_for_Distance_Measurement
[Accessed 2023 November 02].

Multicomp, 2024. *Ultrasonic Ranging Module HC - SR04.* [Online]
[Accessed Marcb 2024].

Multicomp, 2024. *Ultrasonic Ranging Module HC-SR04.* [Online]
Available at: https://www.farnell.com/datasheets/3422740.pdf
[Accessed 18 March 2024].

NanJing Top Power ASIC Corp, 2024. *TP4056 1A Standalone Linear Li-lon Battery Charger with Thermal.* [Online]
Available at: https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Prototyping/TP4056.pdf
[Accessed 21 March 2024].

NASA, 2019. *S H O R T C O U R S E O N L I T H I U M - I O N B A T T E R I E S :.* [Online]
Available at: https://ntrs.nasa.gov/api/citations/20190030819/downloads/20190030819.pdf
[Accessed 9 March 2024].

NHS UK, 2021. *Blindness and vision loss.* [Online]
Available at: https://www.nhs.uk/conditions/vision-loss/
[Accessed 10 February 2024].

Ofcom, 2015. *Cellular Networks and Technology.* [Online]
Available at: https://www.ofcom.org.uk/spectrum/information/cellular-networks-and-technology
[Accessed 15 March 2024].

Ofcom, 2024. *3G and 2G switch-off.* [Online]
Available at: https://www.ofcom.org.uk/phones-telecoms-and-internet/information-for-industry/policy/2g-and-3g-switch-off
[Accessed 21 March 2024].

PennState College of Earth and Mineral Sciences, 2024. *The Satellite Clock.* [Online]
Available at: https://www.e-

education.psu.edu/geog862/node/1714#:~:text=The%20weakness%20of%20gravity%20makes,in%20GPS%20receivers%20on%20earth.
[Accessed 14 March 2024].

PicAxe, n.d. *DF Player Mini.* [Online]
Available at: https://picaxe.com/docs/spe033.pdf
[Accessed 8 November 2023].

Ramos, A. N. C. a. M., 2016. *Obstacle detection using a 2D LIDAR system for an Autonomous Vehicle.* Penang, Malaysia, 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), pp. 441-445.

Shoval, S. a. U. I. a. B. J., 2001. Computerized Obstacle Avoidance Systems for the Blind and Visually Impaired. *Journal of Logic and Computation - LOGCOM,* Volume 978-0-8493-0140-7.

Sight and Sound Technology, 2024. *WeWALK Smart Cane.* [Online]
Available at: https://www.sightandsound.co.uk/product/wewalk/
[Accessed 10 February 2024].

SIMCom, 2013. *https://grobotronics.com/images/companies/1/HC-SR04Users_Manual.pdf.* [Online]
Available at: https://www.makerhero.com/img/files/download/Datasheet_SIM800L.pdf
[Accessed 19 March 2024].

Sparkfun, 2024. *Ultrasonic Ranging Module HC - SR04.* [Online]
Available at: https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf
[Accessed 11 February 2024].

Tarulescu, R. a. S. Ţ., 2012. DISTANCE MEASUREMENTS WITH ULTRASONIC SENSOR DT020-1. *ANNALS OF THE ORADEA UNIVERSITY. Fascicle of Management and Technological Engineering.,* Volume XXI (XI).

The International Engineering Consortium, 2024. *Global System for Mobile Communication (GSM).* [Online]
Available at: https://www.uky.edu/~jclark/mas355/GSM.PDF
[Accessed 15 March 2024].

The University of Pittsburgh, 2024. *What is Vision Impairment?.* [Online]
Available at: https://ophthalmology.pitt.edu/vision-impairment/what-vision-impairment
[Accessed 10 February 2024].

ThePiHut, 2024. *Infrared Proximity Sensor.* [Online]
Available at: https://thepihut.com/products/infrared-proximity-sensor?variant=39648645906627&currency=GBP&utm_medium=product_sync&utm_source=google&utm_content=sag_organic&utm_campaign=sag_organic&gad_source=1&gclid=CjwKCAiA_aGuBhACEiwAly57MZlKEjzk8aeVjj7umC9OSkRxHOsNx
[Accessed 11 February 2024].

TMobile, 2024. *GSM vs. CDMA: What You Need To Know About Phone Bands.* [Online]
Available at: https://www.t-mobile.com/dialed-in/wireless/gsm-vs-cdma-what-you-need-to-know-about-phone-bands
[Accessed 15 March 2024].

Tufts University, 2022. *Obstacle Detection Methods.* [Online]
Available at: https://sites.tufts.edu/eeseniordesignhandbook/files/2022/05/Skantar_TechNotes.pdf
[Accessed 10 February 2024].

U.S Department of Defense, 2018. *What on Earth is the Global Positioning System?.* [Online]
Available at: https://www.defense.gov/News/Feature-Stories/story/Article/1674004/what-on-earth-is-the-global-positioning-system/
[Accessed 14 March 2024].

u-blox, 2011. *NEO-6 u-blox 6 GPS Modules.* [Online]
Available at: https://content.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_%28GPS.G6-HW-09005%29.pdf
[Accessed 20 March 2024].

UltraCane, 2024. *UltraCane.* [Online]
Available at: https://www.ultracane.com/ultracanecat/ultracane
[Accessed 10 February 2024].

Ultracane, n.d. *The UltraCane - an award winning primary mobility aid.* [Online]
Available at: https://www.ultracane.com/about_the_ultracane
[Accessed 16 March 2024].

Ultracane, n.d. *Ultracane.* [Online]
Available at: https://www.ultracane.com/ultracanecat/ultracane
[Accessed 16 March 2024].

University of California, Los Angelos, 2024. *The Global System for Mobile (GSM).* [Online]
Available at: https://web.cs.ucla.edu/classes/fall03/cs218/slides/GSM.pdf
[Accessed 15 March 2024].

University of Pittsburgh, 2011. *Global System for Mobile (GSM).* [Online]
Available at: https://sites.pitt.edu/~dtipper/2700/2700_Slides5K.pdf
[Accessed 15 March 2024].

University of Roma, 2008. *TDMA, FDMA, and CDMA.* [Online]
Available at: http://acts.ing.uniroma1.it/courses/comelet/Slides/20071217_TEL_lecture_2.pdf
[Accessed 12 March 2024].

V. Kunta, C. T. a. U. S., 2020. Multi-Functional Blind Stick for Visually Impaired People. *2020 5th International Conference on Communication and Electronics Systems (ICCES),* pp. 895-899.

WeWALK, 2020. *How it All Began.* [Online]
Available at: https://wewalk.io/en/about/
[Accessed 16 March 2024].

WeWALK, 2020. *The Perfect Pairing for Ultimate Visually Impaired Mobility.* [Online]
Available at: https://wewalk.io/en/product/
[Accessed 16 March 2024].

Yadwinder Singh, L. K., 2017. Obstacle Detection Techniques in Outdoor Environment: Process, Study and Analysis. *International Journal of Image, Graphics and Signal Processing,* Volume 5, p. 19.

Appendix

Appendix I (Interim Report)

Smart Stick for the impaired for Obstacle Avoidance

Student Name: Hamza Afzal

Student ID: 30052265

Supervisor: Ben Mehenni

University of South Wales, Treforest, Cardiff

Introduction (Interim Report)

We will be making Automatic Guidance system for the impaired which has the ability to avoid obstacles. The scope is to create such a device which does not weigh a lot as well as is mobile so the user can carry it anywhere. In addition to this the goal is to make it in the minimum possible budget.

The core of our project is the Ultra Sonic Sensing technology. It is how we will be able to sense if there is an obstacle or not. This will be backed by an Arduino Nano. It has an ATmega 328 microcontroller (Arduino, n.d.) which should be perfect to handle our task. One of our aim is to make this thing as compact and clean as possible, that is also one of the reason why Arduino Nano is our first choice. In Addition to this, it has the required number of pins, so going with another board might've been inefficient as we wouldn't be using all the pins as well as it would be price along it being bigger in size than the Arduino Nano. We can also take a 5V and 3.3V supply from Arduino. But we suspect that it won't be able to supply enough current as we require more furthermore this might cause overheating. Hence the solution to this problem is that we use an external power supply. For the moment, we decided that we will go with rechargeable 4 Ni-Mh [ADD CAPACITY].

Furthermore, we have a water sensor in our order form. We will be ordering a water sensor, but it will be decided whether to use it or not after we are done with the prototyping. The purpose of it is to detect puddle but we are suspicious of how good the Ultrasonic Sensor for is detecting a water puddle. Lastly, our smart blind stick will be equipped with DF Mini MP3 Player which will output audio depending on the condition to guide the user. The output will be to an earphone which will be wired

Background Research of the project (Interim Report)

In this section we will be discussing the thoughts we went through and justifying the decisions we have taken along with the respective options we had available. We will also go through the objections we had on these options and why didn't we choose them.

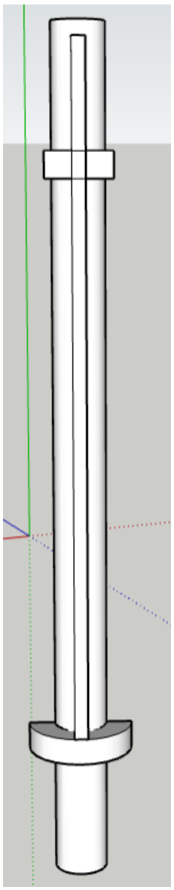
First of all, we wanted to use a microprogramming or a prototyping kit which is versatile. Arduino was our number one choice because it already has all the libraries established to work with the sensors (Arduino, n.d.). We did have Raspberry Pi as an option as well, but we lack experience in working with Raspberry Pi hence lack of knowledge as well which made us hesitant to use it. We chose Arduino Nano specifically because it fulfilled our needs such as the required number of pins. If we went for a bigger board, we'd have extra pin as well as we'd have to pay extra for it. It is tiny in size so it'd help us keep our project compact . We also considered the idea of using the processor standalone, but we'd need to do extra work to program it e.g. installing bootloader and require extra components e.g. crystal oscillator. Therefore Arduino Nano was the best choice.

Next, Ultra Sonic sensor were chosen after thorough research. We considered all options available. These included LiDar, Camera and Infrared sensors. LiDar and Camera are expensive systems so they were out of the equation. Infrared Sensors don't work well if surface properties of objects are unknown (Mohammad, n.d.) although they are cheap compared to Ultra Sonic Sensors. The datasheet for the ultrasonic sensor do tell us that their measurement angle of around 30 degrees (Cytron Technologies Sdn. Bhd, n.d.).

We needed something which could intimidate the user that there is an obstacle hence turn right or turn left. We could do this using a buzzer. An example could be that 3 beeps of the buzzer represent an obstacle at left and 2 beeps represent an obstacle on the right. But that will make the user feel awkward in the public moreover this might disturb people who are near the user. Another consideration was using vibration motor modules which generate vibration, and idea was to use 3 of them, for front, right and left. But it's obvious that constant or periodic vibrations might be really irritating for a user. Furthermore they might also create difficulty in controlling the stick, as stick is a light object. During our research we came across a module called DF Mini MP3 player. This device stores your audios using SD card and we can get the desired audio output when a condition is matched. This voice is transmitted through wired speaker or handsfree. This is a much more suitable solution, no one will get disturbed around the user as there will be no beep sounds. The user will feel much more comfortable because there will be no vibrations. Furthermore, we will be able to keep the project much cleaner in appearance as we will not be using 3 vibration motors. We had other options for audio playback modules like ISD1820, but it stores only one audio and ISD1932 which is not available online, apparently discontinued.

In addition to this we will be using a water sensor. The idea behind this is to detect puddles. Puddle is also considered as an obstacle as it causes hinderance in movement of a person. Water sensor obviously detects if there is water or not. The issue is that how well this idea works practically. A doubt in mind we have is if a person is moving too fast, the sensor might not be able to detect that there is water or not. Even if does will the processor will be fast enough to process it and pass it on to the MP3 player for an output. That is why we have ordered a water sensor as well along with other components.

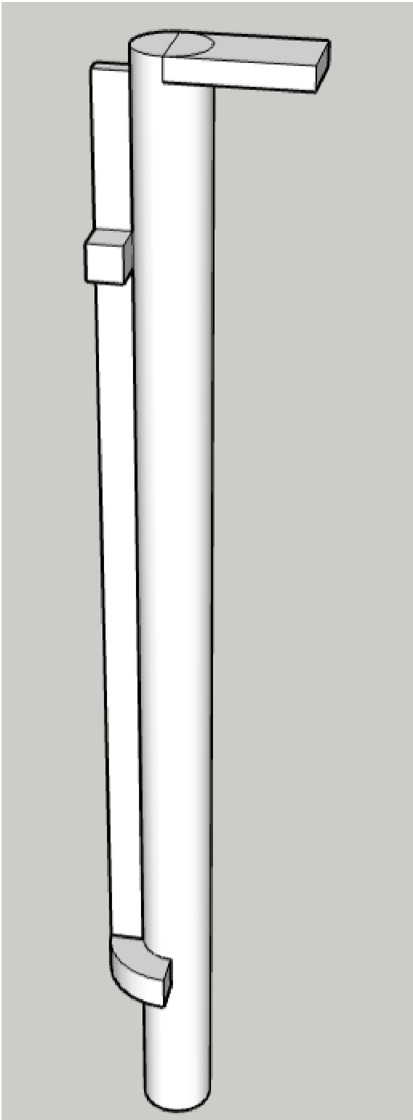
Proposed 3D Model (Interim Report)



Front View

Figure 1 shows the Front view of the stick. Disclaimer: The model is for showcase only; actual dimensions and appearance will differ.

At the top we will have the PCB. We will run a narrow pipe in the middle or some sort of casing to cover the wires. Along with the PCB, we will have one ultrasonic sensor as well. At the bottom we will have two ultrasonic sensors along with the water sensor underneath them. Ultrasonic sensors will be at angle to maximize the coverage. The original idea was to put ultrasonic sensor in a casing but this will totally depend on how that effects the performance of the sensor. We plan to go with a 3D printed casing for all the components.



Side View

The figure above depicts the side view of the proposed model.

Diagrams(Interim Report)

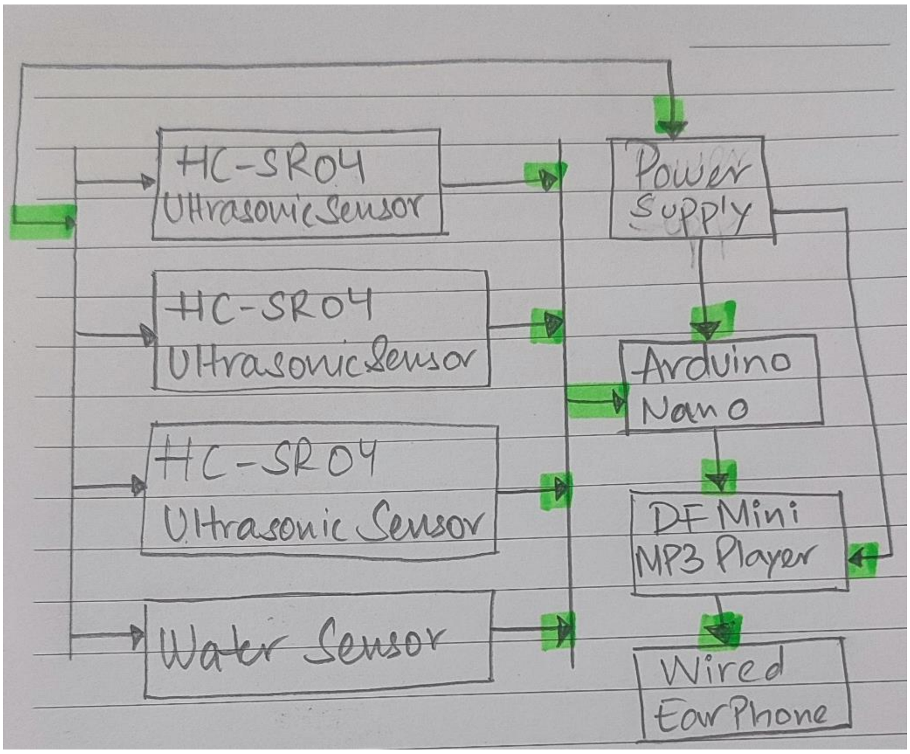
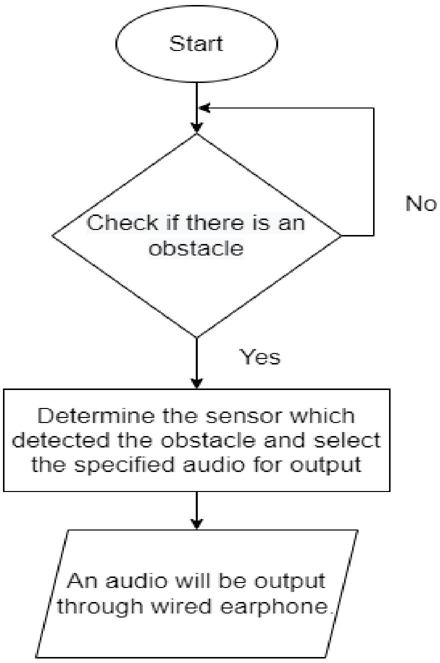


Figure 40, Block Diagram

The diagram, Figure 3, illustrates how the components will be connected. We will have one common power supply for all the components. Ultrasonic sensors and Water sensor are connected to Arduino. Whereas the DF Mini is connected to the Arduino but for output purpose as it acts as a bridge between the earphones and Arduino.



Flowchart on the left displays the algorithm which we will be implementing. It will check if there is an obstacle. If there is none, it will check again. But if there is one it will determine where it is. Then it will signal the DF Mini to send an output via Wired Earphones. The output will contain guidance e.g turn slight right or turn slight left.

Program Flowchart

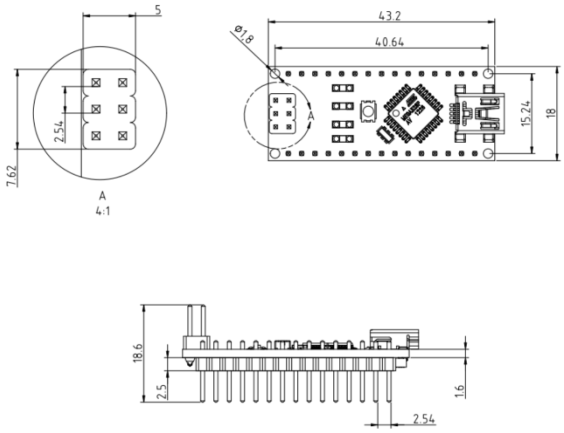
Current Voltage Specification Table (Interim Report)

Name	Voltage	Current
Arduino	5V	19mA (Arduino, n.d.)
Water Sensor	5V	20mA
Ultrasonic Sensor	5V	15mA (Cytron Technologies Sdn. Bhd, n.d.)
DF Mini	5V	20mA (PicAxe, n.d.)

Current Voltage Specification

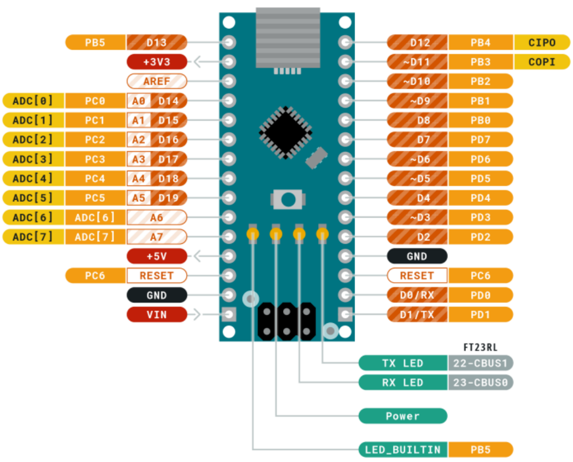
Hardware (Interim Report)

1. Arduino



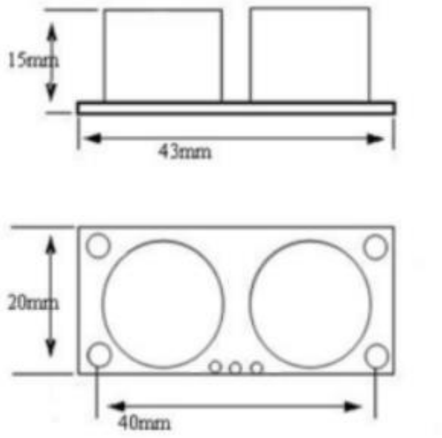
Mechanical dimensions of Arduino Nano

Figure 41, Mechanical Dimensions of Arduino (Arduino, n.d.)

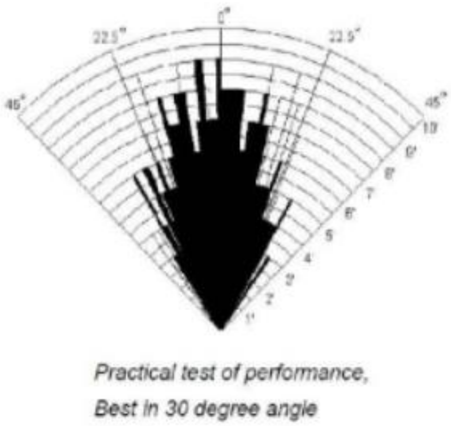


Arduino Pin Out Diagram (Arduino, n.d.)

2. Ultrasonic Sensor

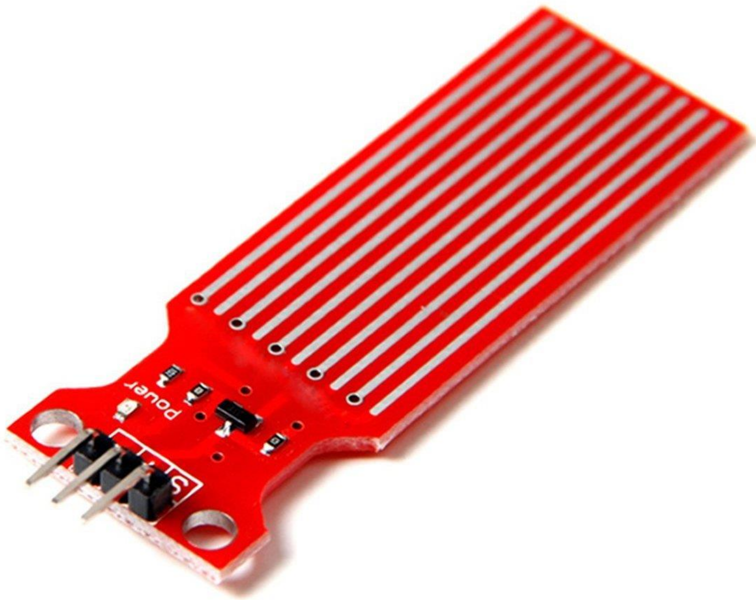


Mechanical Dimensions of HC-SR04 (Cytron Technologies Sdn. Bhd, n.d.)



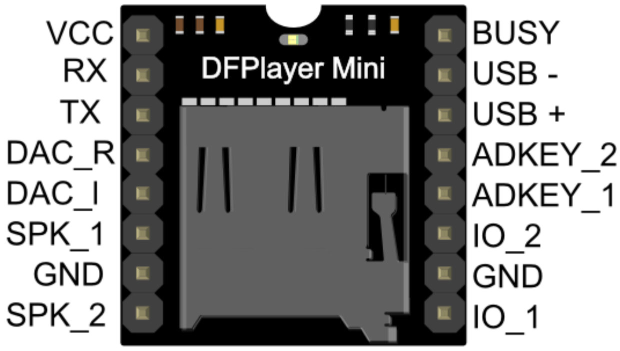
Object Measurement at a Distance and Angle (Cytron Technologies Sdn. Bhd, n.d.)

3. Water Sensor



Water Sensor

4. DF Mini MP3 Player



, DF Mini MP3 Player (PicAxe, n.d.)

Risk Assessment (Interim Report)

For our risk assessment we have determined that there are two specific categories of risks, Generic and Specific.

- Generic Risk Assessment.
  These are risk which are considered common and can happen any day. Some of these risks are as follows:

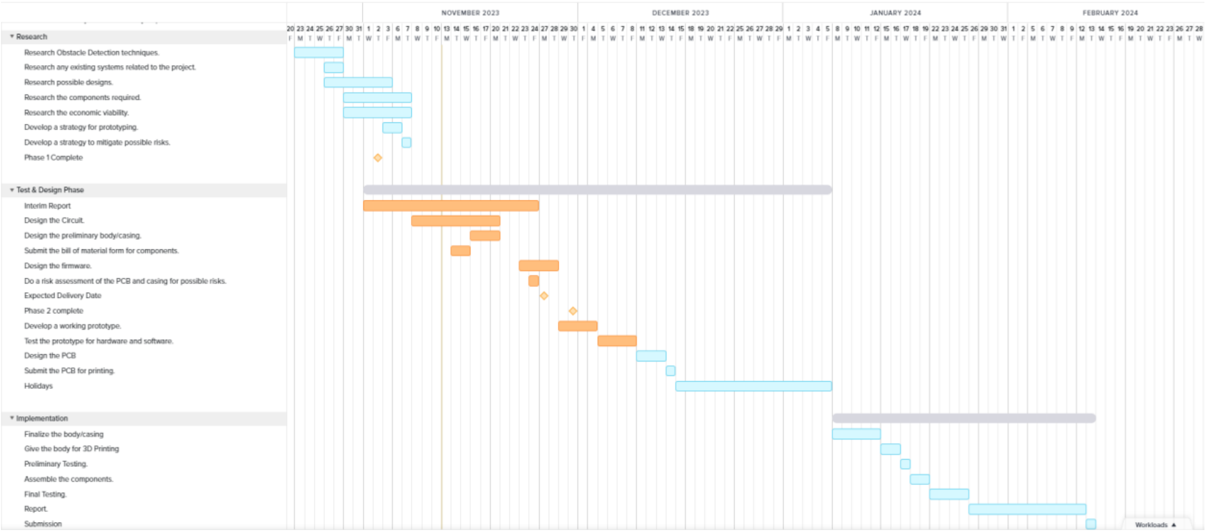
  1. Theft of Materials
  2. Power Outage
  3. Tools can go faulty.
  4. Delay in components' delivery.
  5. An epidemic
  6. Delays due to emergency e.g Medical Reasons, Sick Leave

- Specific Risks
  These risks are the ones which are related to my project and components only and can occur anytime but not in another sort of project. A few are described below:

Risk No.	Description of Risk	Probability of the Risk	Effect on the project	Actions to mitigate the risk

1	Short Circuit leading to fire. Can Happen due to faulty components or mishandling of wires or battery/power source.	High	Can delay the project. New Components might have to be ordered.	Check each component individually. Before turning on power supply check all the connections thoroughly.
2	Burn from Soldering. If soldering iron is not handled with care.	Low	Can damage the effected body parts. Project delay or slow down until healed.	Handle the soldering iron with care. Wear safety gloves.
3	Data Loss. Loss of Data e.g documentation and code	Low	Will reset the project if all data is wiped. Will have to redo all the lost data.	Store data on a cloud storage as well. Keep a copy of data in pendrive when possible.
4	Connectivity Issue. Due to faulty and malfunctioned copper tracks.	Low	Need to do extra soldering using jumper wires. Will make the project look unclean.	Solder with care. Don't apply too much heat on the PCB.
5	Heat. Due to high data processing demand among components.	Low	Can damage/Malfunction components hence will need to reorder.	Proper ventilation for components. Use heatsinks where possible.
6	Sensing Limitation. Sensing can be affected due to various conditions such as atmospheric effects. Sensor by default are limited in sensing data in certain conditions.	Medium	Outliers in the data.	Use the sensor according to the manufacturers guideline and given optimum conditions.

Gantt Chart



Gantt Chart

The above is our Gantt Chart. We are almost in line with what we have planned. We will follow it as far as there are no external interventions described in the risk assessment as well as other than that.

Current State of the Project (Interim Report)

Current State of the project is that we are ready to submit the Order Form. We will be waiting for the delivery of the components. Meanwhile, we are designing the circuit as well as the program to run the Arduino on. Once we get the components, we will design a prototype hence the circuit and the program are necessary. After that we will design the chassis as we will have the dimensions. This way we will also be able to design the PCB because we will have the circuits as well as the dimensions. Once we are done with the prototype, we will send another order form the PCB which we should be able to receive in January. So, we should be done with assembly at the end of January.

References (Interim Report)

Aircom International, 2002. *https://sites.pitt.edu/~dtipper/2700/2700_Slides5K.pdf.* [Online]
Available at: https://www.pce-fet.com/common/library/books/50/7377_1-GSMOverviewAircom.pdf
[Accessed 15 March 2024].

Arduino, 2019. *Serial.available().* [Online]
Available at: https://www.arduino.cc/reference/en/language/functions/communication/serial/available/
[Accessed 2024 March 2024].

Arduino, 2024. *Arduino Nano 33 IoT.* [Online]
Available at: https://store.arduino.cc/products/arduino-nano-33-iot
[Accessed 17 March 2024].

Arduino, 2024. *Arduino Nano.* [Online]
Available at: https://store.arduino.cc/products/arduino-nano
[Accessed 1 03 2024].

Arduino, 2024. *Arduino Nano 33 BLE.* [Online]
Available at: https://store.arduino.cc/products/arduino-nano-33-ble
[Accessed 17 March 2024].

Arduino, 2024. *Arduino Nano 33 BLE Sense.* [Online]
Available at: https://store-usa.arduino.cc/products/arduino-nano-33-ble-sense
[Accessed 17 March 2024].

Arduino, 2024. *Powering Alternatives for Arduino Boards.* [Online]
Available at: https://docs.arduino.cc/learn/electronics/power-pins/
[Accessed 17 March 2024].

Arduino, 2024. *Product Reference Manual.* [Online]
Available at: https://docs.arduino.cc/resources/datasheets/A000005-datasheet.pdf
[Accessed 17 March 2024].

Arduino, 2024. *Serial.begin().* [Online]
Available at: https://www.arduino.cc/reference/en/language/functions/communication/serial/begin/
[Accessed 18 March 2024].

Arduino, 2024. *Softwareserial.* [Online]
Available at: https://docs.arduino.cc/learn/built-in-libraries/software-serial/
[Accessed 22 March 2024].

Arduino, 2024. *TinyGPS.* [Online]
Available at: https://www.arduino.cc/reference/en/libraries/tinygps/
[Accessed 22 03 2024].

Arduino, n.d. *Arduino Nano.* [Online]
Available at: https://store.arduino.cc/products/arduino-nano
[Accessed 02 November 2023].

Arduino, n.d. *Arduino Nano Data Sheet.* [Online]
Available at: https://docs.arduino.cc/resources/datasheets/A000005-datasheet.pdf
[Accessed 04 November 2023].

Arduino, n.d. *Libraries.* [Online]
Available at: https://www.arduino.cc/reference/en/libraries/
[Accessed 02 November 2023].

Bhatta, N. M. P., 2017. RADAR and its applications. *International Journal of Circuit Theory and Applications,* Volume 10, pp. 1-9.

Components 101, 2021. *SIM800L GSM Module.* [Online]
Available at: https://components101.com/wireless/sim800l-gsm-module-pinout-datasheet-equivalent-circuit-specs
[Accessed 19 March 2024].

Cytron Technologies Sdn. Bhd, n.d. *Data Sheet for Ultra Sonic sensor.* [Online]
Available at:
https://www.researchgate.net/publication/258883212_Using_Ultrasonic_and_Infrared_Sensors_for_Distance_Measurement
[Accessed 3 November 2023].

Cytron Technologies, 2013. *User's Manual.* [Online]
Available at: https://grobotronics.com/images/companies/1/HC-SR04Users_Manual.pdf
[Accessed 18 March 2024].

DFRobot, 2024. *SKU:TEL0124.* [Online]
Available at: https://wiki.dfrobot.com/SIM7600CE-T_4G(LTE)_Shield_V1.0_SKU_TEL0124
[Accessed 21 March 2024].

Discant, A. a. R. A. a. R. C. a. B. A., 2007. *Sensors for Obstacle Detection - A Survey.* s.l., IEEE, pp. 100-105.

Electronic Clinic, n.d. *Time-of-Flight ranging Sensor.* [Online]
Available at: https://www.electroniclinic.com/wp-content/uploads/2021/06/TOF10120-Datasheet.pdf
[Accessed 11 February 2024].

Farnell, 2024. *BANNER ENGINEERING QT50R-US-AFH.* [Online]
Available at: https://uk.farnell.com/banner-engineering/qt50r-us-afh/radar-sensor-24m-time-of-flight/dp/3872615
[Accessed 10 March 2024].

Farnell, 2024. *DFROBOT SEN0259.* [Online]
Available at: https://nl.farnell.com/en-NL/dfrobot/sen0259/lidar-laser-range-sensor-tfmini/dp/3769959
[Accessed 10 March 2024].

Farnell, 2024. *DFROBOT SEN0395.* [Online]
Available at: https://uk.farnell.com/dfrobot/sen0395/mmwave-radar-board-arduino-board/dp/3879712?cjevent=0c781db2c8ce11ee810600c00a18b8f9&cjdata=MXxZfDB8WXww&CMP=AFC-CJ-UK-7947610&gross_price=true&source=CJ
[Accessed 11 February 2024].

Farnell, 2024. *MULTICOMP PRO PSG04176.* [Online]
Available at: https://uk.farnell.com/multicomp-pro/psg04176/ultrasonic-distance-sensor/dp/4162010?gad_source=1&gclid=Cj0KCQjwqpSwBhClARIsADlZ_TmB_w2hFE6JMCdxHqL1Vu6zICbWVoY1NM8OeR4zmuVDmVa5nIy8-

TUaAhuJEALw_wcB&mckv=_dc|pcrid||plid||kword||match||slid||product|4162010|p
[Accessed 10 March 2024].

Federal Aviation Administration, 2022. *Satellite Navigation - GPS - How It Works.* [Online]
Available at:
https://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/navservices/gnss/gps/howitworks#:~:text=To%20accomplish%20this%2C%20each%20of,that%20provide%20extremely%20accurate%20time.
[Accessed 14 March 2024].

Gbenga, D. S. A. a. A. A., 2017. Smart walking stick for visually impaired people using ultrasonic sensors and Arduino. *International journal of engineering and technology,* 9(5), pp. 3435-3447.

GeeksforGeeks, 2020. *Difference between FDMA and TDMA.* [Online]
Available at: https://www.geeksforgeeks.org/difference-between-fdma-and-tdma/
[Accessed 15 March 2024].

GISGeography, 2024. *How GPS Receivers Work – Trilateration vs Triangulation.* [Online]
Available at: https://gisgeography.com/trilateration-triangulation-gps/
[Accessed 14 March 2024].

GPS.GOV, 2022. *Timing.* [Online]
Available at:
https://www.gps.gov/applications/timing/#:~:text=Each%20GPS%20satellite%20contains%20multiple,receiver%20to%20the%20atomic%20clocks.
[Accessed 14 March 2024].

Gupta, M. P. A. a. A. R., 2018. *Smart Stick for the Blind and Visually Impaired People.* Coimbatore, s.n., pp. 542-545.

L. Koval, J. V. P. B., 2016. Distance Measuring by Ultrasonic Sensor. *International Federation of Automatic Control,* 49(25), pp. 153-158.

Massachusetts institute of technology, 2020. *Batteries & Energy Storage.* [Online]
Available at: https://ocw.mit.edu/courses/2-60j-fundamentals-of-advanced-energy-conversion-spring-2020/ed4b85c28d16b5cc2dd0d2635a5723f4_MIT2_60s20_lec11.pdf
[Accessed 9 March 2024].

Microchip, 2015. *8-bit AVR Microcontroller with 32K Bytes In-System.* [Online]
Available at: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf
[Accessed 13 March 2024].

Mohammad, T., n.d. *Using Ultrasonic and Infrared Sensors for.* [Online]
Available at:
https://www.researchgate.net/publication/258883212_Using_Ultrasonic_and_Infrared_Sensors_for_Distance_Measurement
[Accessed 2023 November 02].

Multicomp, 2024. *Ultrasonic Ranging Module HC - SR04.* [Online]
[Accessed Marcb 2024].

Multicomp, 2024. *Ultrasonic Ranging Module HC-SR04.* [Online]
Available at: https://www.farnell.com/datasheets/3422740.pdf
[Accessed 18 March 2024].

NanJing Top Power ASIC Corp, 2024. *TP4056 1A Standalone Linear Li-lon Battery Charger with Thermal.* [Online]
Available at: https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Prototyping/TP4056.pdf
[Accessed 21 March 2024].

NASA, 2019. *S H O R T C O U R S E O N L I T H I U M - I O N B A T T E R I E S :.* [Online]
Available at: https://ntrs.nasa.gov/api/citations/20190030819/downloads/20190030819.pdf
[Accessed 9 March 2024].

NHS UK, 2021. *Blindness and vision loss.* [Online]
Available at: https://www.nhs.uk/conditions/vision-loss/
[Accessed 10 February 2024].

Ofcom, 2015. *Cellular Networks and Technology.* [Online]
Available at: https://www.ofcom.org.uk/spectrum/information/cellular-networks-and-technology
[Accessed 15 March 2024].

Ofcom, 2024. *3G and 2G switch-off.* [Online]
Available at: https://www.ofcom.org.uk/phones-telecoms-and-internet/information-for-industry/policy/2g-and-3g-switch-off
[Accessed 21 March 2024].

PennState College of Earth and Mineral Sciences, 2024. *The Satellite Clock.* [Online]
Available at: https://www.e-education.psu.edu/geog862/node/1714#:~:text=The%20weakness%20of%20gravity%20makes,in%20GPS%20receivers%20on%20earth.
[Accessed 14 March 2024].

PicAxe, n.d. *DF Player Mini.* [Online]
Available at: https://picaxe.com/docs/spe033.pdf
[Accessed 8 November 2023].

Ramos, A. N. C. a. M., 2016. *Obstacle detection using a 2D LIDAR system for an Autonomous Vehicle.* Penang, Malaysia, 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), pp. 441-445.

Shoval, S. a. U. I. a. B. J., 2001. Computerized Obstacle Avoidance Systems for the Blind and Visually Impaired. *Journal of Logic and Computation - LOGCOM,* Volume 978-0-8493-0140-7.

Sight and Sound Technology, 2024. *WeWALK Smart Cane.* [Online]
Available at: https://www.sightandsound.co.uk/product/wewalk/
[Accessed 10 February 2024].

SIMCom, 2013. *https://grobotronics.com/images/companies/1/HC-SR04Users_Manual.pdf.* [Online]
Available at: https://www.makerhero.com/img/files/download/Datasheet_SIM800L.pdf
[Accessed 19 March 2024].

Sparkfun, 2024. *Ultrasonic Ranging Module HC - SR04.* [Online]
Available at: https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf
[Accessed 11 February 2024].

Tarulescu, R. a. S. Ţ., 2012. DISTANCE MEASUREMENTS WITH ULTRASONIC SENSOR DT020-1. *ANNALS OF THE ORADEA UNIVERSITY. Fascicle of Management and Technological Engineering.,* Volume XXI (XI).

The International Engineering Consortium, 2024. *Global System for Mobile Communication (GSM).* [Online]
Available at: https://www.uky.edu/~jclark/mas355/GSM.PDF
[Accessed 15 March 2024].

The University of Pittsburgh, 2024. *What is Vision Impairment?.* [Online]
Available at: https://ophthalmology.pitt.edu/vision-impairment/what-vision-impairment
[Accessed 10 February 2024].

ThePiHut, 2024. *Infrared Proximity Sensor.* [Online]
Available at: https://thepihut.com/products/infrared-proximity-sensor?variant=39648645906627&currency=GBP&utm_medium=product_sync&utm_source=google&utm_content=sag_organic&utm_campaign=sag_organic&gad_source=1&gclid=CjwKCAiA_aGuBhACEiwAly57MZlKEjzk8aeVjj7umC9OSkRxHOsNx
[Accessed 11 February 2024].

TMobile, 2024. *GSM vs. CDMA: What You Need To Know About Phone Bands.* [Online]
Available at: https://www.t-mobile.com/dialed-in/wireless/gsm-vs-cdma-what-you-need-to-know-about-phone-bands
[Accessed 15 March 2024].

Tufts University, 2022. *Obstacle Detection Methods.* [Online]
Available at: https://sites.tufts.edu/eeseniordesignhandbook/files/2022/05/Skantar_TechNotes.pdf
[Accessed 10 February 2024].

U.S Department of Defense, 2018. *What on Earth is the Global Positioning System?.* [Online]
Available at: https://www.defense.gov/News/Feature-Stories/story/Article/1674004/what-on-earth-is-the-global-positioning-system/
[Accessed 14 March 2024].

u-blox, 2011. *NEO-6 u-blox 6 GPS Modules.* [Online]
Available at: https://content.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_%28GPS.G6-HW-09005%29.pdf
[Accessed 20 March 2024].

UltraCane, 2024. *UltraCane.* [Online]
Available at: https://www.ultracane.com/ultracanecat/ultracane
[Accessed 10 February 2024].

Ultracane, n.d. *The UltraCane - an award winning primary mobility aid.* [Online]
Available at: https://www.ultracane.com/about_the_ultracane
[Accessed 16 March 2024].

Ultracane, n.d. *Ultracane.* [Online]
Available at: https://www.ultracane.com/ultracanecat/ultracane
[Accessed 16 March 2024].

University of California, Los Angelos, 2024. *The Global System for Mobile (GSM).* [Online]
Available at: https://web.cs.ucla.edu/classes/fall03/cs218/slides/GSM.pdf
[Accessed 15 March 2024].

University of Pittsburgh, 2011. *Global System for Mobile (GSM).* [Online]
Available at: https://sites.pitt.edu/~dtipper/2700/2700_Slides5K.pdf
[Accessed 15 March 2024].

University of Roma, 2008. *TDMA, FDMA, and CDMA.* [Online]
Available at: http://acts.ing.uniroma1.it/courses/comelet/Slides/20071217_TEL_lecture_2.pdf
[Accessed 12 March 2024].

V. Kunta, C. T. a. U. S., 2020. Multi-Functional Blind Stick for Visually Impaired People. *2020 5th International Conference on Communication and Electronics Systems (ICCES),* pp. 895-899.

WeWALK, 2020. *How it All Began.* [Online]
Available at: https://wewalk.io/en/about/
[Accessed 16 March 2024].

WeWALK, 2020. *The Perfect Pairing for Ultimate Visually Impaired Mobility.* [Online]
Available at: https://wewalk.io/en/product/
[Accessed 16 March 2024].

Yadwinder Singh, L. K., 2017. Obstacle Detection Techniques in Outdoor Environment: Process, Study and Analysis. *International Journal of Image, Graphics and Signal Processing,* Volume 5, p. 19.

Table of Figures (Interim Report)

University of South Wales
Prifysgol De Cymru

Appendix II

Arduino Final Code (final v9)

final_v9.ino

```
#include <TinyGPS++.h>
#include <SoftwareSerial.h>

TinyGPSPlus gps;

SoftwareSerial neo6m(4, 5); //Tx, Rx for neo6m
SoftwareSerial sim800l(7, 6); // Tx, Rx for SIM800L module

#define BUTTON_PIN 8 // Pin where the button is connected
unsigned long previousMillis1 = 0;
unsigned long previousMillis2 = 0;
unsigned long lastBlinkTime = 0;
const long interval1 = 500; // Interval for LED1 blink (in milliseconds)
const long interval2 = 700; // Interval for LED2 blink (in milliseconds)
const int ledPin2 = 10; // pin for Led 1
const int ledPin1 = 11;// pin for Led 2
const int buzzerPin = 9; // Pin for buzzer
const int trigPin1 = 13; // Ultrasonic Sensors Trig and Echo pins
const int echoPin1 = 12; //**
const int trigPin2 = 3;//**
const int echoPin2 = 2;//**
boolean buttonPressed = false; //Button State initially not pressed
int emergency_state=1; // Initializing Emergency State as 1
int ledState1 = LOW; // Initial state for LED1
int ledState2 = LOW; // Initial state for LED2

boolean checkResponse(String expectedResponse, unsigned long timeout = 1000) {
  boolean responseFlag = false; // flag for response
  String serverResponse = ""; // string variable to store response
  unsigned long startTime;
  for (startTime = millis(); (millis() - startTime) < timeout;) { // millis function used to capture
    while (sim800l.available()) { //check's if sim800l is available or not
      serverResponse = sim800l.readString(); // stores the response from the server
      if (serverResponse.indexOf(expectedResponse) > 0) { // condition sets the flag to true if posi
        responseFlag = true;
        goto OUTSIDE;
      }
    }
  }
  OUTSIDE:
  if (serverResponse != "") { Serial.println(serverResponse); } // if the response is something else
  return responseFlag;
}
boolean sendCommand(String cmd, String expectedResponse, int responseTimeout, int attempts) {
  Retry:

  for (int i = 1; i <= attempts; i++) { // for loop to send command
    sim800l.println(cmd);
    if (checkResponse(expectedResponse, responseTimeout) == true) { // if the condition is true the
      break;
    }
    else {
      Serial.print(".");
    }
  }

}
```

```
void sendSMS(String number, String msg) {
  Serial.println("Calling Sim800l");
  sendCommand("AT+CMGS=\"" + String(number) + "\"", ">", 1000, 5); // this will try send
  sim800l.print(msg);
  sim800l.write(26); // ASCII code for Ctrl+Z to send the message
  buttonPressed = true; // mark the button as pressed to avoid sending multiple SMS
  Serial.println("Button Pressed");
}

// Buzzer function for the right sensor
void buzzerRight(int distance) {

  if (distance < 30) {
    tone(buzzerPin, 1000); // Play a tone
    delay(150); // delay required for beeps
  } else if (distance < 45) {
    tone(buzzerPin, 500); // Play a different tone
    delay(150);// delay required for beeps
  } else {
    noTone(buzzerPin); // Turn off the buzzer if the distance is more the defined ranges
  }
}
// Buzzer function for the left sensor
void buzzerLeft(int distance) {
  if (distance < 30) {
    tone(buzzerPin, 1100); // Play a tone
    delay(500);
  } else if (distance < 45) {
    tone(buzzerPin, 528); // Play a different t
    delay(500);
  } else {
    noTone(buzzerPin); // Turn off the buzzer i
  }
}

void startupBuzzer(){
   // Unique sound sequence with the buzzer
  int notes[] = {262, 330, 392, 523}; // Freque
  int duration = 200; // Duration of each note

  for (int i = 0; i < 4; i++) { // for loop to
    tone(buzzerPin, notes[i], duration); // pla
    delay(duration); // pause between notes
    noTone(buzzerPin); // turn off the buzzer
    delay(50); // delay for pause
  }
}
```

```
void ultrasonic(int trigPin, int echoPin, int buzzerPin) {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  long duration = pulseIn(echoPin, HIGH); // Reads the echoPin,
  int distance = duration * 0.034 / 2; // Calculating distance
  Serial.print("Distance from the object = ");
  Serial.print(distance);
  Serial.println(" cm");

  if (trigPin == trigPin1 && echoPin == echoPin1) {
    // If the right side sensor, produce a sound relative to th
    buzzerRight(distance);
  } else if (trigPin == trigPin2 && echoPin == echoPin2) {
    // If the left side sensor, produce a different sound patte
    buzzerLeft(distance);
  }
}
```

```
void gps_location(){
  neo6m.begin(9600);
  neo6m.listen();

  unsigned long startMillis = millis();
  while (millis() - startMillis < 1000) { // Wait for 1 sec
    while (neo6m.available()) { // checks if the neo6m is a
      gps.encode(neo6m.read()); // reads data from neo6m if
      if (gps.location.isUpdated()) { // if location is new
        Serial.print("Latitude= ");
        Serial.print(gps.location.lat(), 6); // translate t
        Serial.print(" Longitude= ");
        Serial.println(gps.location.lng(), 6); // translate
        return; // exit the function once location is updat
        neo6m.end(); // end communication with the Module
      }
    }
  }
  Serial.println("No GPS data received within 1 second");
}
```

University of South Wales Prifysgol De Cymru

```
void blinkLED(){

  unsigned long currentMillis = millis(); // Get the curr

  // Blink LED1
  if (currentMillis - previousMillis1 >= interval1) {
    previousMillis1 = currentMillis; // Save the last tim
    if (ledState1 == LOW) {
      digitalWrite(ledPin1, HIGH); // Turn LED1 on
      ledState1 = HIGH; // Update LED1 state
    } else {
      digitalWrite(ledPin1, LOW); // Turn LED1 off
      ledState1 = LOW; // Update LED1 state
    }
  }

  // Blink LED2
  if (currentMillis - previousMillis2 >= interval2) {
    previousMillis2 = currentMillis; // Save the last tim
    if (ledState2 == LOW) {
      digitalWrite(ledPin2, HIGH); // Turn LED2 on
      ledState2 = HIGH; // Update LED2 state
    } else {
      digitalWrite(ledPin2, LOW); // Turn LED2 off
      ledState2 = LOW; // Update LED2 state
    }
  }
}

```

```
void setup(){
  Serial.begin(9600); // Initialise the c
  // Declare the modes for all different
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  pinMode(trigPin1, OUTPUT);
  pinMode(echoPin1, INPUT);
  pinMode(trigPin2, OUTPUT);
  pinMode(echoPin2, INPUT);
  startupBuzzer(); // call this function
}
```

```
void loop() {
  // Check for obstacles while checking emergency state
  ultrasonic(trigPin1, echoPin1, buzzerPin); // Calculates distance from the 1st Sensor
  ultrasonic(trigPin2, echoPin2, buzzerPin); // Calculates distance from the 2nd Sensor

  // Check button press for sending emergency message
  if (digitalRead(BUTTON_PIN) == HIGH && !buttonPressed) {
    emergency_state = 0; // in emergency state, state is changed from 1 to 0
    gps_location(); // Call GPS function for location
    sim800l.begin(9600); // Start comm with SIM800L
    sim800l.listen(); // Call the SIM800L module
    String message = "HELP ME!!! https://www.google.com/maps?q=" + String(gps.location.lat(), 6) + "," + String(gps.location.lng(), 6)
    Serial.println(message);
    sendSMS("447888196873", message); // Send SMS to this number
    lastBlinkTime = millis(); // Capture the time for counting later for LEDs
    buttonPressed = true; // Mark the button as pressed
  }

    // Check emergency state
  if (emergency_state != 1) {
    sim800l.end();
    buttonPressed = false;
    if (millis() - lastBlinkTime < 300000) {  // Check if 5 minutes have passed since the LEDs started blinking
      blinkLED(); // Blink LED function to blink LEDs
    } else {
      digitalWrite(ledPin1, LOW); // Turn off LEDs
      digitalWrite(ledPin2, LOW);
      emergency_state = 1; // Back to normal state
    }

  }

}
```