NAVIGATION STUDY BY USING RFID NETWORK FOR VISUALLY IMPAIRED PERSON

JO JO SAM WAI LUN

The report is submitted in Partial Fulfilment of Requirements for the Degree of Bachelor in Mechatronic Engineering

Faculty of Electrical Engineering
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2013/2014

“I hereby declared that I have read through this report entitled “Navigation Study by Using RFID Network for Visually Impaired Person” and found that it has comply he partial fulfilment for awarding the degree of Bachelor of Mechatronic Engineering”

Signature : ....

Supervisor’s Name : ....

Date : ....

I declare that this report entitle "Navigation Study by Using RFID Network for Visually Impaired Person" is the result of my own research except as cited in the references. The report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : ....

Name : ....

Date : ....

To my beloved Mother and Father

ACKNOWLEDGEMENT

Firstly, I would like to take this opportunity to express my highest gratitude to my supervisor, Engr. Anuar bin Mohammed Kassim for his guidance, monitoring and constant motivation throughout final year project. Without his support and encouragement, this project would not have been presented same as here. Under his guidance and supervision, a lot of great knowledge, ideas and inspiration .I had obtained to complete this thesis. I would like to thank and appreciate for the engineering lesson that had delivered from all the lecturers.

Many thanks also all the lecturers UTeM has given lecture to me and provide guidance and suggestion for the realization of this project. Also special thanks to technician that allows me to use laboratory equipment and provides components to ensure a successful project. Any cooperation given would be greatly appreciated.

Last but not least, I thanks to my parents and family for support and encouragement through the final year project. And thanks to my friends, course mate and seniors for their constant blessing and encouragement to guide me. Also special thanks to UTeM for giving me opportunity to pursue my engineering course. Once again I express my sincere thanks to all involved either directly or indirectly to this project. Thank you.

ABSTRACT

Navigation is the most concern for the visually impaired person. It is very hard for them to travel independently from one to another destination without the proper navigation tools. The advanced technology developed nowadays help visually impaired people travelling safely and independently. There are many ways to help visually impaired people such as guide dog, white cane and tactile paving. RFID technology is introduced in this project to guide visually impaired people at indoor. In order to provide an efficient and user-friendly navigation tools, it is proposed to design and develop a navigation device using RFID network to guide visually impaired person walking properly via verbal navigation. The idea of positioning or localization with compass and direction guiding though voice commands is implemented in this project in order to guide navigation such as moving straight forward and turn corner. The A* algorithm is then implemented in this project to determine the shortest path. The navigation device is connected to the laptop through wireless connection in order to obtain the travel guidance from the map system. The real-time test will be held inside the FKE building of Universiti Teknikal Malaysia Melaka (UTeM). The research focuses on the calibration of the compass and relocates the visually impaired back to normal route before them out of the direction. Data analysis will be done during the evaluation of the navigation device and improvement will be made to improve the performance of the device. The blind navigation device developed is able to guide the visually impaired people to reach destination with the shortest path. This project is beneficial to visually impaired person because the navigation device developed with sound navigation will be integrated with the electronic cane as it helps them to have a better, safer and comfortable travel.

Keyword
RFID(Radio Frequency Identification) , Navigation , visually impaired people

ABSTRAK

Navigasi adalah kebimbangan yang paling untuk orang yang cacat penglihatan. Ia amat sukar bagi mereka untuk perjalanan bebas dari satu ke destinasi lain tanpa alat navigasi yang sesuai. Teknologi canggih yang dibangunkan pada hari ini membantu orang cacat penglihatan perjalanan dengan selamat dan bebas. Terdapat banyak cara untuk membantu orang cacat penglihatan seperti panduan anjing , tongkat putih dan jalan sentuhan. Teknologi RFID diperkenalkan dalam projek ini untuk membimbing orang cacat penglihatan di dalam bangunan. Dalam usaha untuk menyediakan satu alat navigasi yang cekap dan mesra pengguna, ia adalah dicadangkan untuk mereka bentuk dan membangunkan peranti navigasi menggunakan rangkaian RFID untuk membimbing orang cacat penglihatan berjalan dengan betul melalui navigasi lisan. Idea mengenalpasti kedudukan atau penyetempatan dengan kompas dan arah membimbing melalui arahan suara dilaksanakan dalam projek ini untuk membimbing navigasi seperti bergerak lurus ke hadapan dan belok kiri kanan. A * algoritma kemudiannya dilaksanakan dalam projek ini untuk menentukan laluan terpendek . Navigasi peranti disambungkan kepada komputer riba melalui sambungan tanpa wayar untuk mendapatkan panduan perjalanan dari sistem peta. Ujian sebenar - masa akan diadakan di dalam bangunan FKE di Universiti Teknikal Malaysia Melaka (UTeM). Kajian ini memberi tumpuan kepada penentukuran kompas dan menempatkan semula orang yang cacat penglihatan kembali dengan jalan biasa sebelum mereka keluar dari arahan itu. Analisis data akan dilakukan semasa penilaian navigasi peranti dan penambahbaikan akan dibuat untuk meningkatkan prestasi peranti. Navigasi peranti dapat memberi petunjuk kepada orang cacat penglihatan untuk sampai ke destinasi dengan jalan yang singkat. Projek ini memberi manfaat kepada orang cacat penglihatan kerana peranti navigasi dibangunkan dengan navigasi bunyi akan disepadukan dengan tongkat elektronik kerana ia membantu mereka dalam perjalanan yang lebih baik , lebih selamat dan selesa.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Motivation

According to the World Health Organization, there are 285 million people are visually impaired worldwide; 39 million are blind while 246 million have low vision. The causes of blindness are due to eye-sighted problem, 43%, disambiguation of eyes, 33% and glaucoma, 2%. There are two categories of people who are at risk, they are people aged 50 and over and children below aged 15. About 65% of all people who are visually impaired are aged 50 and older; this group comprises about 20% of the world's population [1].

Outdoor and indoor pedestrian mobility are very difficult and dangerous for visually impaired people. They usually rely on a cane or guide dog to assist them in reaching desired destination safely. However, this approach is successful if only if the path to the destination is already familiar or known by the blind people or the guide dog. It becomes difficult once the destination is new and the orientation is difficult especially the environment is not designed for blind people [2].

Public locations such as crowded shopping malls, airports, train stations, and bus stations can be difficult to navigate and can be become disorienting for those with visual impairments. These public spaces contain various sensory distractions such as traffic noise and other people. For a visually impaired people, it can be difficult to determine what direction to travel without some form of guidance. Navigating through unfamiliar public locations has long been source of difficulty for the blind [3].There is some contributions done by researchers to design the navigation system to help visually impaired and blind people for travelling purpose.

S.Wills and ***S.Helal et al.*** proposed RFID tag grid on the navigation and location determination system to guide the visually impaired people during travelling. RFID tags that contain the coordinate's information in being installed on the floor, therefore user can walk on the footpath easily [4]. ***Yuriko***, ***Yoshiaki*** and ***Kenji et al.*** developed navigation systems that help the visually impaired people to walk independently at indoor. The navigation system uses RFID reader and color sensor attached on the white cane to detect the color navigation line and the RFID tags [5].

Furthermore, ***Andrew*** and ***Satish et al.*** proposed navigational system using RFID network. They designed the system using PDA device stored in pocket that have RF reader antennas to receive the navigation information from the RFID tags that are mounted at the ceilings or wall. Until now, the research of the navigation system for the visually impaired people is more focusing on the indoor navigation. The uses of GPS system is not available implemented in the indoor navigation due to some reasons like signal interference and blocking [6].

1.2 Problem Statement

Nowadays although visually impaired people have white cane as their main travelling tools, sometimes they will lost their direction at the end. They face problems like avoiding obstacles, sensing environment and determining current location. They tend to seek help from the pedestrian and this will make them feel uncomfortable from keep asking for direction. Besides, the tactile of the white cane is work as obstacle detection but it not fully guides the proper navigation as it does not give feedback to the user. For example, the blind people might turn left and right or might be go far away from the tactile paving and hence they lost their way back to previous location.

Visually impaired people need to navigate in their nearby environment like normal sighted peoples. We are as normal sighted humans do need visual cues or detailed source of information during our travel. In this case, the way-finding becomes challenges and difficult for visually impaired people. Even though they have conventional navigational aids, these assistances are not feasible or possible all the times. For example, obstacles like door and elevator nearby with direction and distance or presence of human in motion [8].

The problem of indoor navigation remains unsolved. The safety issue addressed by the navigation aids is less useful for indoors. Traveling by visual impaired people during indoor tend to be dangerous because of its homogenous environment. Besides searching or touch –sensing for certain unique, the visually impaired people have to count doorways and intersections to differentiate between the indoor facilities such as offices or doorways and this seems extremely difficult for them and time-consuming [7].

1.3 Objectives

The purpose of this project is to develop a navigation device by using RFID network for visually impaired people. This can be addressed by the following objectives:

1. To design and develop navigation device using RFID network to guide visually impaired people for travelling.
2. To navigate the visually impaired people about the walking and turning.
3. To evaluate the performance of the developed navigation device in terms of lab test accuracy and validity of algorithm.

1.4 Scope

1. The device developed is only a lab prototype to guide navigation for visually impaired people.
2. The device is designed for visually impaired people to walk independently only for indoor purpose.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This literature review of the project focuses on the types of navigation device used in helping the visually impaired person on walking independently and path finding algorithm for the navigation device. There is a need to build a navigation device to guide the navigation for the visually impaired through RFID network due to many difficulties faced by them. There are few parts will be discussed in this literature review which are theoretical background, performance criteria and design challenges, navigation device for visually impaired person, comparison indoor navigation system, navigation study of path finding algorithm, comparison of path finding algorithm. Conclusion are made at the end of this chapter about what navigation device is chosen and the path finding algorithm is studied to solve the problem when the user is out of guided path in order to lead the visually impaired people a safe and comfortable travelling.

2.2 Theoretical Background

Navigation is the important needs in daily life. Independent travel of visual impaired person is difficult. A navigation device designed should be consistent and accurate providing navigation is both indoor and outdoor navigation. Outdoor navigation system successfully guides the user however indoor navigation applications are still improving the consistency factor.

Global National Satellite Systems (GNSS) or GPS cannot be reached into the buildings because these signals are disturbed or scattered by walls and roofs of the buildings. Indoor navigation faces the following challenges:

- Location identification of the user
- Navigation once the initial position is determined
- Map creation for indoor travelling

- Adaptability with environment changes

In order to get the solution of navigation, there is a need to reproduce GPS signals inside the buildings using the outdoor navigation techniques. These techniques can be classified into three techniques; kinematic navigation techniques, wireless navigation techniques and visual navigation techniques [9].

2.2.1 Kinematic navigation techniques

This technique use smartphone built-in sensors like accelerometer, gyroscope and magnetometers to determine position, orientation and velocity of a moving project using kinematic equations. Dead reckoning algorithm is used for navigating a moving device relative to unknown start point, orientation and velocity [9].

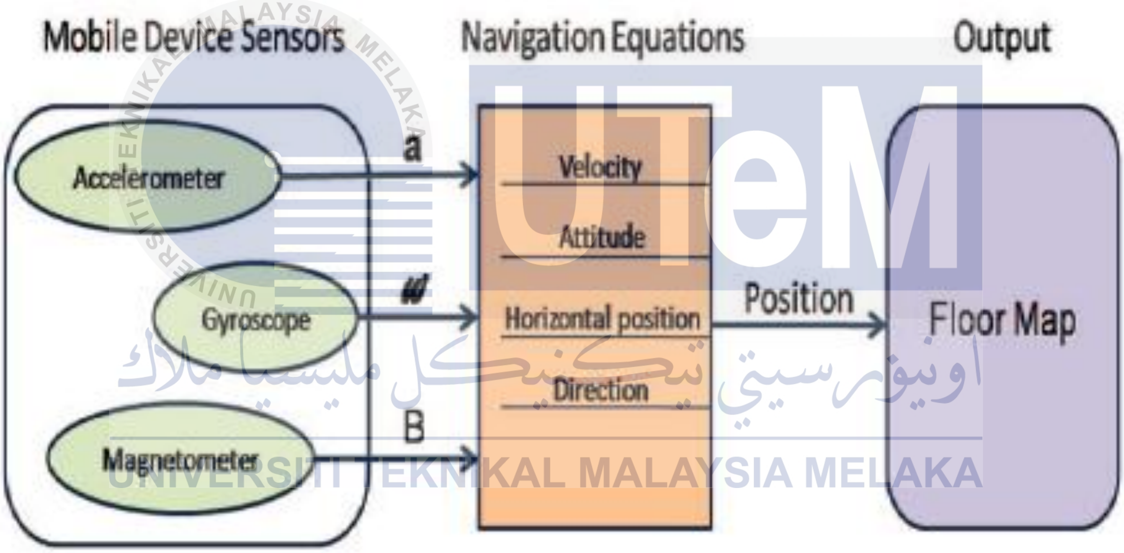


Figure 2.1: Kinematic navigation-based indoor navigation

2.2.2 Wireless navigation techniques

This techniques use radio waves or light waves to determine user's position. This approach is same as mobile phones is capable of transmitting and receiving signals inside the buildings based on Bluetooth, RFID or Wi-Fi. Principles like Cell of Origin, Least Square or Triangulation is used to determine the position of the user while Trilateration, Bancroft's method is used to track the position. Direction of the user can be detected by methods such as Received signal strength (RSS), Angle of Arrival (AoA), Time of Arrival and Time Difference of Arrival (TDoA) [9].

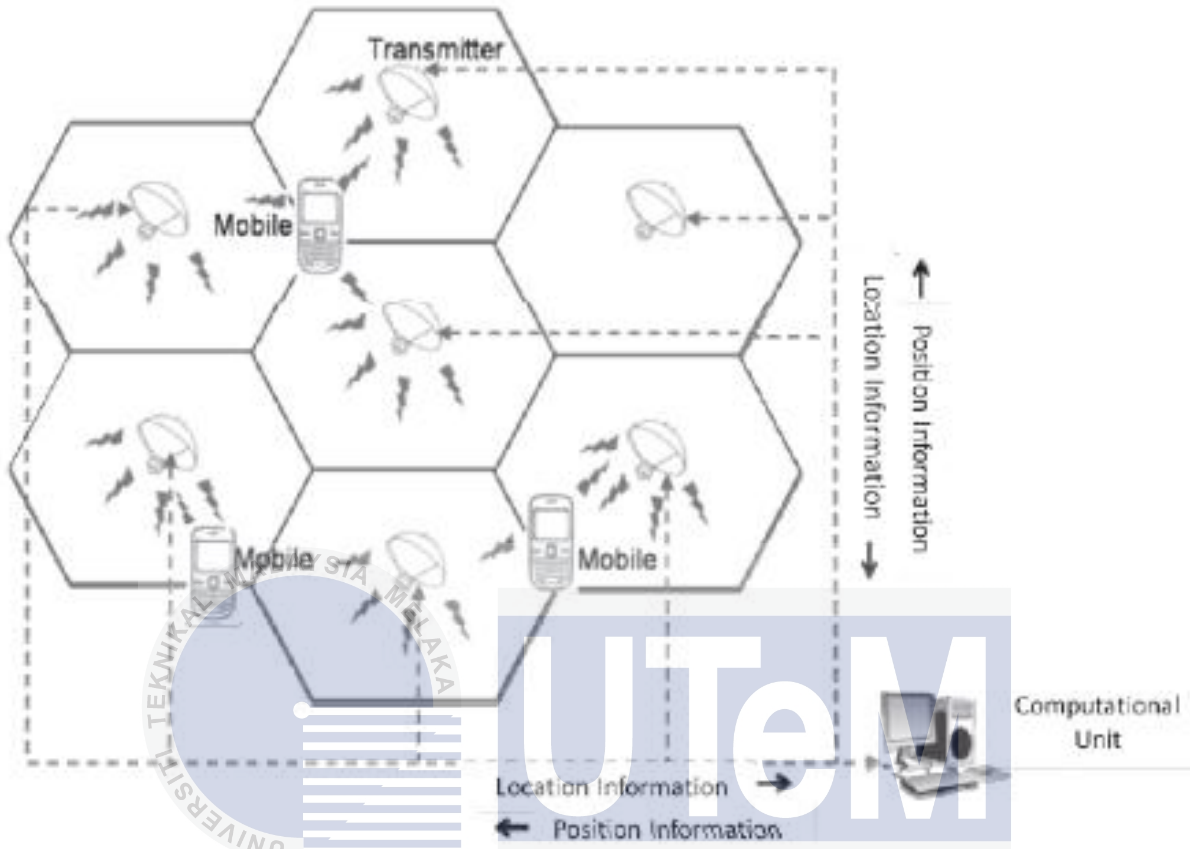


Figure 2.2: Wireless techniques for indoor navigation

2.2.3 Visual navigation techniques

This technique is like camera-based positioning techniques focuses on determining the objects which are visible. Location information can be in encoded form (barcode or QR-code).These code contains the information regarding the location and made this technique more accuracy in positioning. The image as information can be digitized and checked for location matches in database [9].

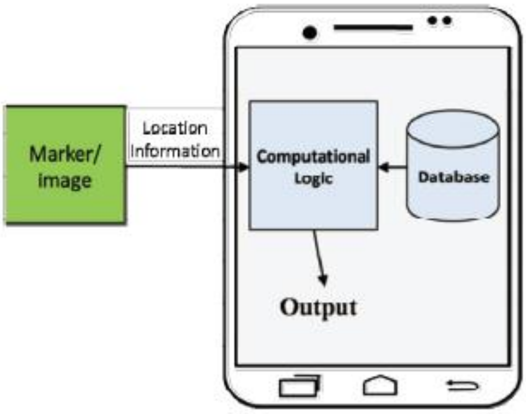


Figure 2.3: Visual navigation techniques in mobiles

2.3 Performance criteria

The development of indoor navigation application is growing and thus performance criteria need to be determined to design useful blind navigation system. The design of the indoor navigation system is remains unsolved. In order to improve future design, the performance criteria has been divided into many parts; positioning accuracy, robustness, integration with changing environment and information received by the user [7].

Table 2.1: Summary of performance criteria

	Considerations	Requirements
Accuracy	Information received from tags to navigation device	• Deliver accuracy for the perceptual range from primary device. Accuracy of less than 2 meters.
Robustness	The correctness of the information received by user	• Provide a guaranteed error to the readings.
Integration with environment	Feasibility of external infrastructural	• Extensible of the navigation device
Information received by user	Distances and directions	• Distances should be expressed in meters. • Bearing should be

		expressed with the reference to the user's current heading.

2.3.1 Design challenges

The design challenges for the blind navigation device are real-time guidance, portability, power limitations, appropriate interface, and continuous availability, no dependence on infrastructure, low cost solution and minimal training [7].

There are several designs to help visually impaired people to achieve self-independence travelling at indoor environment. Location technology such as RFID, IrDA, Bluetooth or WIFI has been developed to help them travel during indoors with contextual information or sound navigation.

The use of GPS device can help to guide the visually impaired person in an outdoor environment. This voice-based system guides them with the shortest route to the destination. Since GPS cannot function in indoor spaces, other researchers present the solution by using infrared technology standard (IrDA) that works as sensors to determine the indoor location. The Drishti system is the combination of GPS for outdoor navigation and ultrasonic sensor for indoor navigation. One of the problems about GPS is the error with the measurement taken especially inside the tall buildings. Furthermore, the user has to carry two ultrasonic sensors to receive signals that are transmitted from different point in the rooms. With these signals, the server can analyze the data and thus detect the locations. Other than that, grid system that contains RFID tags that can read by RFID reader, and the system will monitor the movements and the path taken by users by WIFI communications [10].

2.4 Consideration for Navigation Device

S.Santhosh, T.Sasipoabha and R. Jeberson has proposed a navigation system named BLI-NAV designed for visually impaired people. GPS receiver and path detector are used in the system for detecting coordination or user's location and thus determine the shortest route to the destination. Voice command given by user in order to start the navigation and the user current location is determined by GIS database from the voice information. Critical path algorithm used to determine the shortest distance from the starting point to end point with the path detector. User can avoid obstacles through voice command while travelling. GIS database not only can store all details of map information but also can adapt to the changes the map information in other cities or countries [11].

M.Choudhury, D.Aguerrevere and A.Barreto proposed the Pocket-Pc based Electronic Travel Aid (ETA) to help visually impaired people to travel at indoor environments. Ultrasonic range sensor is used to detect surrounding obstacles and an electronic compass is used for direction navigation purpose. Pocket –PC will alert user where he or she is near the obstacles. The warning alert audio is played back through stereo headphones so user can perceive surrounding obstacles. This system gave the better results in real time performance and improves the efficiency of unhindered blind travel in indoor environments [12].

Fukusawa and Magatani presented the developed navigation system that can help the visually impaired person to walk independently. This system uses the color sensor to detect the color of navigation line and then informs the user by vibration if they are near the obstacles. The direction identification technique of the system helps to detect a user's walking direction. However, the systems are not working well at indoor environment and costly. Color type sensor that used in this system can be affected by lightning condition where sensor fails to detect light under the dark environment condition [13].

B.Mounir, A.Redjati, M. Fezari and M.Bettayeb designed a navigation device to help blind and visually impaired people for travelling. Microcontroller is used in the navigation device that can produce speech as output. The device guide the user what is the route should be taken. Besides, this aid helps to reduce navigation difficulties and obstacles detection using ultrasounds and vibrators. Stereoscopic sonar system is used to detect the nearest obstacles in this system and then it sends back vibro-tactile feedback to tell the user about the current location [15].

E.Kaiser and M.Lawo presented navigation device to aid visually impaired people travel at indoor and outdoor. Short range laser, initial measurement unit (IMU), portable computer and headphones are used to work together with the system. Position of the pedestrian can be detected and tracked by the system. This system implements Simultaneous Localization and Mapping (SLAM) for this system. User is guided perfectly with route selecting method once the maps are created. The purpose of this designed system is to provide efficient navigation with the white cane [16].

M.Shamsi, M.Qutayri and J.Jeedella presented the implementation of a system that helps blind person navigate independently at indoor environment such as the home. The system provides the localization by using wireless mesh network. Server which is doing the path planning communicates wirelessly with the portable mobile unit. The visually impaired person can commands and receive response from the server via audio signals [17].

S.Chumkamon, P.Tuhaphanthaphiphat, P. Keeratiwintakorn proposed that navigation system using RFID network for the visually impaired person. Shortest distance from the current location to the destination can be found using this system. Besides, the system can help them to find their way back if they lost their ways and recalculate the new path. RFID reader is used to read the RFID tags which are embedded inside the footpath [18].

A.Ganz, S.Gandhi, C.Wilson, G.Mullett presented indoor navigation system named INSIGHT to assist the blind person to travel inside the buildings. The system uses RFID with Bluetooth technology to locate the user inside the building. The PDA based user device interacts with INSIGHT server and provide navigation information through audio form. The zone that user walked is being monitor by the system. The system will notify the user if user travels the wrong direction [19].

Table 2.2: Comparison between indoor navigation systems

Title	Device	Benefits
BLI-NAV Embedded Navigation System for Blind People	GPS receiver and path detector	• Determine the shortest path using critical path algorithm. • User avoids obstacles while travelling. • Voice command through GIS system. • GIS system can store and update map information.
A Pocket-Pc based Navigational Aid for Blind Individuals	Ultrasonic sensors and electric compass	• Generate a warning alert audio when user is near the obstacles. • Users perceive the nearby obstacles.
A navigation system for the visually impaired and intelligent white cane	Color Sensor	• Sense navigation line color • Alert user through vibration
An Ultrasonic Navigation System For Blind People	Microcontroller with speech output, stereoscopic sonar system	• Gives information to user about urban route • Reduce navigation difficulties and obstacles detection using ultrasounds and vibrators
Wearable Navigation System for the Visually Impaired and Blind People	Short range laser, (IMU) inertial measurement unit and computer and headphones.	• User guided perfectly with route selecting method.
Blind Assistant Navigation System	Wireless Mesh Network	• Path planning done by system communicates wirelessly with user. • User can command and receive from server via audio signals.
Blind Navigation System Using RFID for Indoor Environments	RFID-based navigation	• Find shortest path to destination. • Detect the direction lost and recalculate the route.

INSIGHT: RFID and Bluetooth Enabled Automated Space for the Blind and Visually Impaired	RFID with Bluetooth technology	• User interacts with server through audio. • System monitors user walk. • System can identify the wrong way lead to destination.

In conclusion of the consideration of navigation device, RFID-based navigation is chosen for the indoor navigation for this project. RFID (radio frequency identification technology) has the advantages of low cost and high flexible that the visually impaired person can travel on the footpath that contain the RFID tags. The RFID tag reader can read the information stored by the tags.

2.4.1 RFID Tag

RFID tags can be categorized into different types such as passive, active or battery-assisted passive. Tag is read only and it has the unique identification code that is used to access into the database.

Figure 2.4 RFID Tag

2.4.2 RFID Reader

RFID tag reader receives only radio signals from active tags. The reception range of the reader can adjusted from 0.30 – 610m.

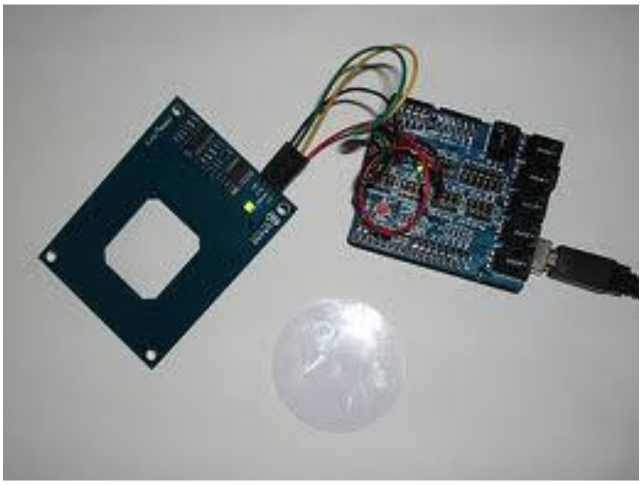


Figure 2.5 RFID Reader

2.5 Navigation Study on Path Planning

R.K Megalingam, A.P Rajendran and D.Dileepkumar proposed LARN Algorithm for indoor automatic navigation system. The navigation system is installed on a wheelchair and it is controlled by joystick and processing system. LARN algorithm is used by this system to travel inside house to determined location. Complete rotation of the wheelchair can used to measure the distance travelled and the angle of orientation is measured using digital compass using this algorithm [20].

K.C Fan and P.C Lui presented a path planning algorithm to solve the path-finding problem in mapped environments. Modified A* algorithm based on depth-first search in setting up the *h'*value is used for path planning strategies. The path planning algorithm is find path from start node to a destination node through graph and chose the best path based on cost function. The performance of this path planning algorithm is evaluated in terms of optimality of results and execution speed. The cost of modified A* algorithm is larger than depth-first search but the time needed is far lesser. The modified A* algorithm is better because it can find good path in a short time [21].

N.A.Vlassis, N.M.Sgouros, G.Efthivoulidis and G.Papakonstantinou proposed global path planning method for autonomous qualitative navigation at indoor. Autonomous qualitative navigation is working based on the changes of sensor behavior between regions in space. The method is based on modified version of the Dijsktra's shortest path algorithm. The algorithm is tested in real-time environment to find the destinations. Qualitative map (Qmap) is used to determine the shortest path from the starting point to the goal point at the end of the algorithm [22].

E.J Gene, M.C Lee, S.G Hsieh and Y.Y Chen proposed Least-Turn Path Algorithm and Minimum-Cost Path Algorithm in the path planning from a source node to a destination node. Two main parameters in path planning of navigation are path length and number of turns. Rectilinear routing algorithm is used to find least-turn path while Kirby's concept and modified Dijkstra's algorithm is introduced for minimum cost algorithm. The shortest path, the least-turn path, the shortest path with least turns and the least-turn path with the shortest length can be found using this minimum-cost path algorithm [23].

Y. Toda and N. Kubota presented topological map generation method for robots. This method is based on Delaunay Triangulation able to detect the allowable space automatically. This method can analyze and generate suitable path and correct moving direction on the detected allowable space. Topological node based on landmark can be converted into tools for indoor navigation [24].

X.W Fu, Z. Liu and X.G Gao introduce path planning algorithm for unmanned aerial vehicle (UAVs) to avoid radar network. Delaunay triangulations graph is constructed based on locations. Initial path is searched by using Dijkstra algorithm. Optimization approach is then used to calculate the vector that UAV needs to navigate to goal position. The results show the path followed this algorithm not only help UAV to avoid the radar network but also achieve the UAV task-required limitations [25].

C. Liu and J.Wu presented face routing in position-based routing. In order to improve the efficiency of position-based routing, message from a node is delivered to another node that has no neighbor close to the destination called local minimum in greedy mode. A virtual small network is constructed by adding the long links to network to reduce the chance of meeting the local minima. SWING is the combination of face routing and virtual force method that recover from local minima. Results show SWING finds shorter route and this method is truly greedy which can works in unfamiliar environment [26].

Table 2.3: Comparison between Paths Planning Algorithm

Title	Method	Benefits
LARN: Implementation of Automatic Navigation in Indoor Navigation for Physically Challenged	LARN Algorithm	• Measure the distance travelled by the wheelchair and digital compass is used to measure the orientation angle.
Solving Find Path Problem in Mapped Environment Using Modified A* Algorithm	Modified A* algorithm	• Find path from start node to the end node using graph and chose the suitable path
Global Path Planning for Autonomous Qualitative Navigation	Dijsktra Algorithm	• Qualitative map (Qmap) is used to determine the shortest distance from the starting point to goal point.
Transportation Network Navigation with Turn Penalties	Minimum-Cost Path Algorithm and Least-Turn Path Algorithm	• Determine the shortest path, the least-turn path, the shortest path with least turns and the least-turn path with the shortest length.
Path Planning using Multi-resolution Map for a Mobile Robot	Delaunay Triangulation	• Analyze and generate allowable path with true direction based on detected allowable space
Path Planning for UAV in radar network area	Dijkstra algorithm	• Search the initial path • Calculate the vector that UAV needs to navigate to goal position
SWING:Small World Iterative Navigation Greedy Routing Protocol in MANETs	Face Routing	• Find Shorter Route in unfamiliar environment

2.5.1 LARN Algorithm

LARN Algorithm that proposed by ***R.K Megalingam, A.P Rajendran and D.Dileepkumar*** [20] is being on the wheelchair for the indoor automatic navigation. The wheelchair has two navigation modes. One is the manual mode which is controlled by joystick while the second is automatic mode. The system monitors the location and updates the information each time the wheelchair passes the grid. The house's floor plan is partitioned into number of small lattice in LARN method. The size of grid is same as the total base size of wheelchair. The wheelchair completes a rotation once passes a grid.

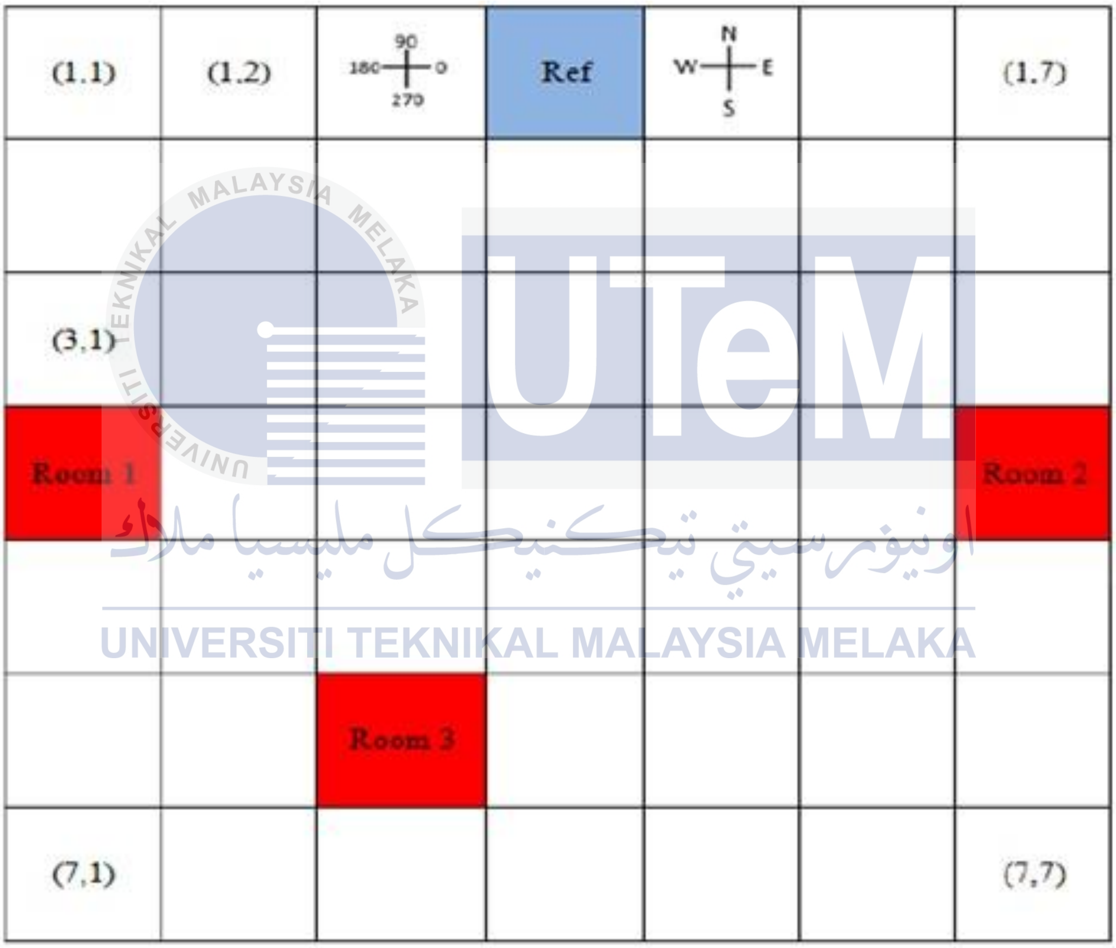


Figure 2.6: Floor Plan

Figure 2.6 shows the floor plan of the house. The blue grid represents the house entrance and it is marked as the reference while the red grid is the destination point. Digital compass is used to determine the direction heading of wheelchair using angle with reference to the magnetic pole. North, South, East and West corresponds to the angle 90 °, 270 °, 0 °and 180 ° respectively.

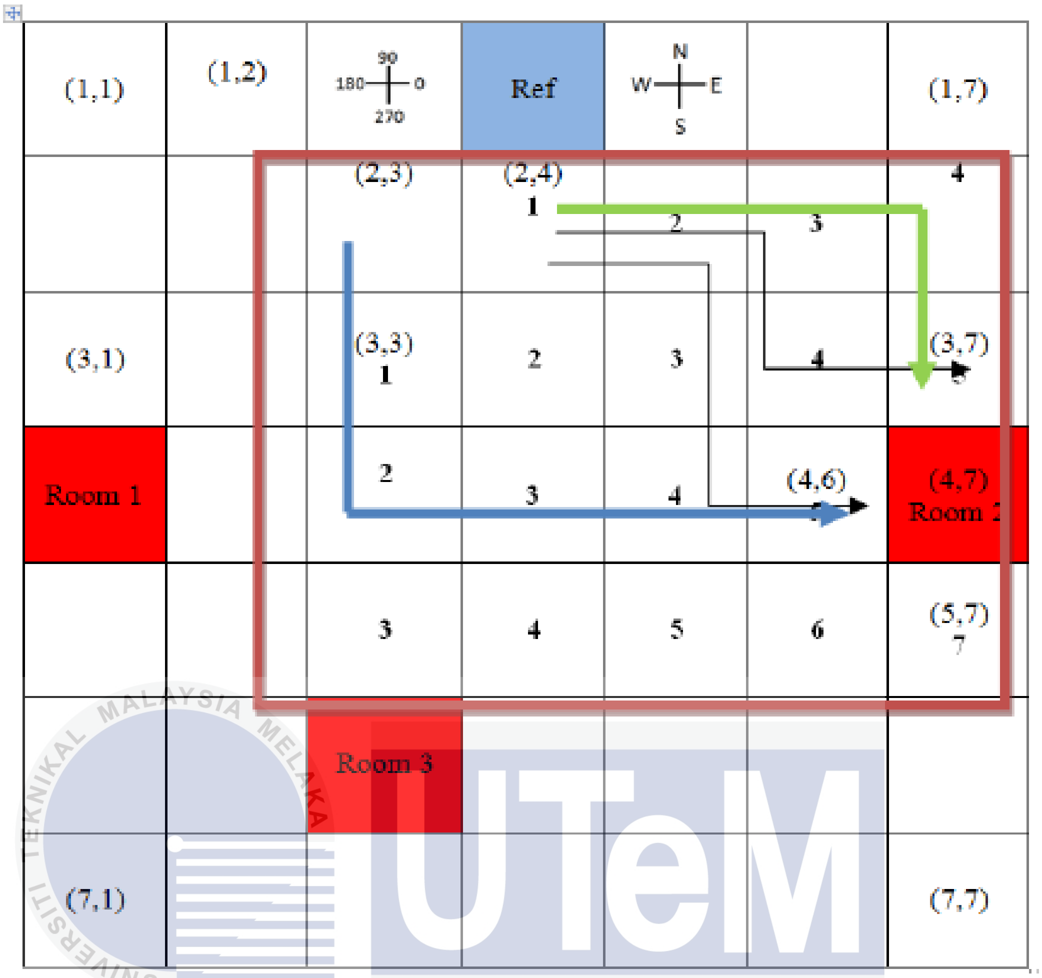
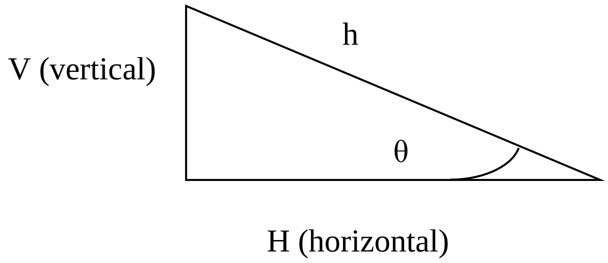


Figure 2.7: Implementation of LARN Algorithm on floor plan

Figure 2.7 shows the implementation of LARN algorithm. After determining the shortest possible paths, it is known that there are 4 ways to reach the destination. However, there are two shortest paths with least number of 90° where the blue line takes only 7 steps while the green line takes only 5 steps.

It is found that angle measured from the present position from the starting point is the multiple of ($\pi/2$) from the experimental results. If the orientation angle is equal to the multiple of ($\pi/2$), the system recalculates the horizontal and vertical distance travelled using right angle triangle. The initial orientation angle as θ and distance travelled as hypotenuse, h. Equations are obtained once the triangle is constructed.



$$V = h \times \sin(\theta) \quad (2.1)$$

$$H = h \times \cos(\theta) \quad (2.2)$$

2.5.2 Topological Map Generation Using Delaunay Triangulation

Y. Toda and N. Kubota proposed the topological map generation based on Delaunay Triangulation for mobile robot. This method is useful in indoor navigation. Various paths can be planned to reach the destination. The advantage of using topological maps is it helps to find the shortest path between non-adjacent nodes [24].

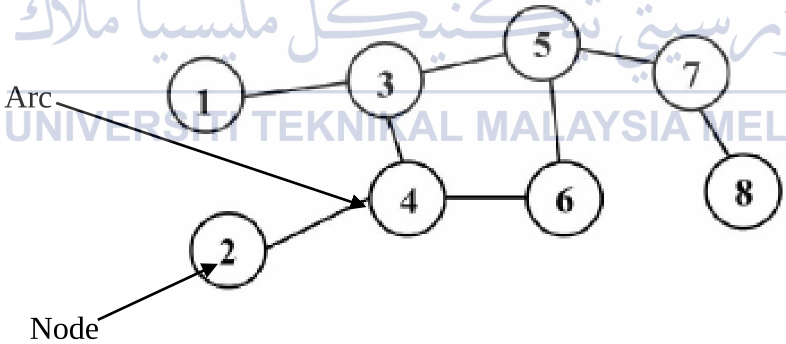


Figure 2.8: Topological map

Figure 2.8 shows the topological map. This map is useful because the elements of the map are almost same as the environment. However, the limitation of this method is landmark recognition. This map cannot be used under unknown environment because the topological sequence will broke. Topological map has nodes like landmarks and the arcs are linked. Nodes like landmarks represent workspace while the arc represents the node’s connection. Labeling of the nodes with numbers defined the action for moving direction is shown as Figure 2.9 below.

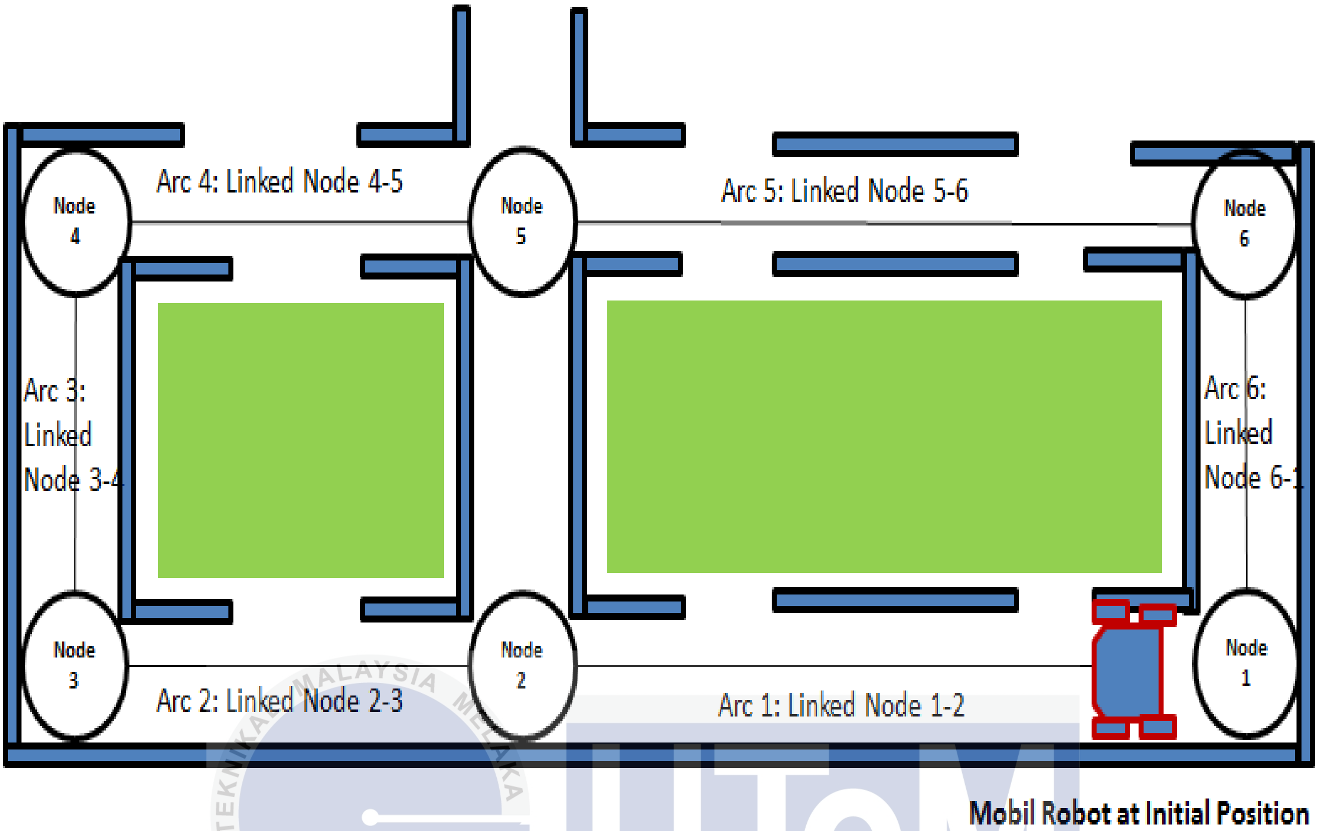


Figure 2.9: Graphical representation of topological map

Initially, the first step of this method uses the laser rangefinder to detect the maximum distance it can be. According to the range of maximum distance R, facilities structure such as dead ends, left/right corners, T-junctions and intersections. The landmark of allowable space within maximum distance R is found by Delaunay triangulation. Delaunay triangulation of a point set is a collection of edges satisfying an "empty circle" property. Allowable workspace detections from Delaunay Triangulation usually are dead-end, left/right corner, T-junction and intersection as shown in Figure 2.10.

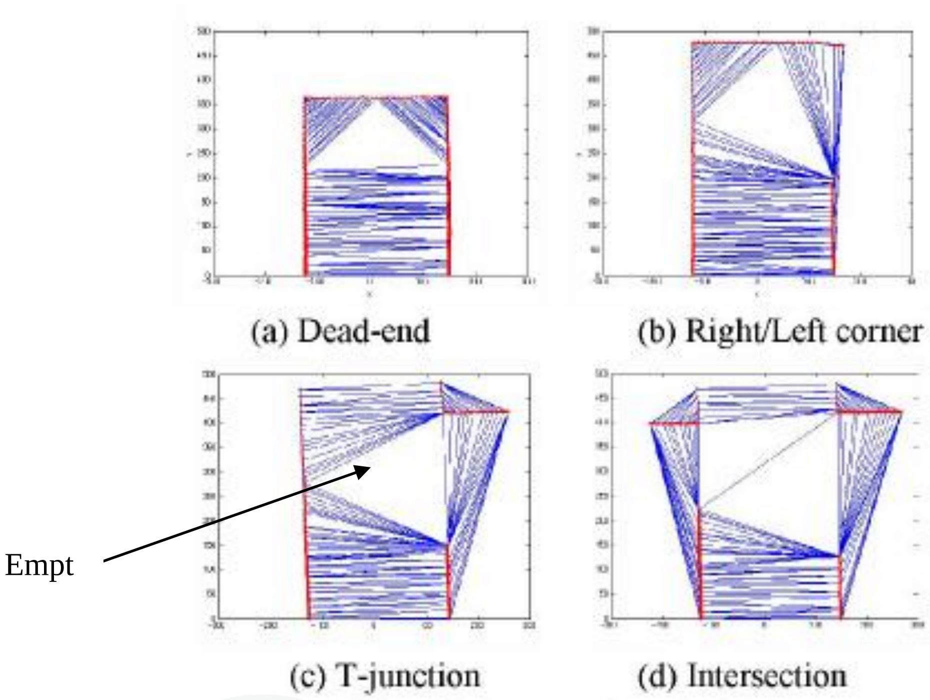


Figure 2.10: Workspace of Delaunay Triangulation

Step 2 is the sides of each Delaunay Triangulation are measure using threshold. Threshold means the minimum workspace width that the robot can move around. The triangle is eliminated if the any sides of a Delaunay triangle do not pass the threshold.

Step 3 is appropriate path with sufficient space is required to move the robot from its current position to the predetermined position. The allowable workspace for the robot is determined by the midpoint of Delaunay triangle side. The allowable path is the middle point of each sides of triangle as shown in Figure 2.11. It is required to eliminate the path that does not have the landmark triangle.

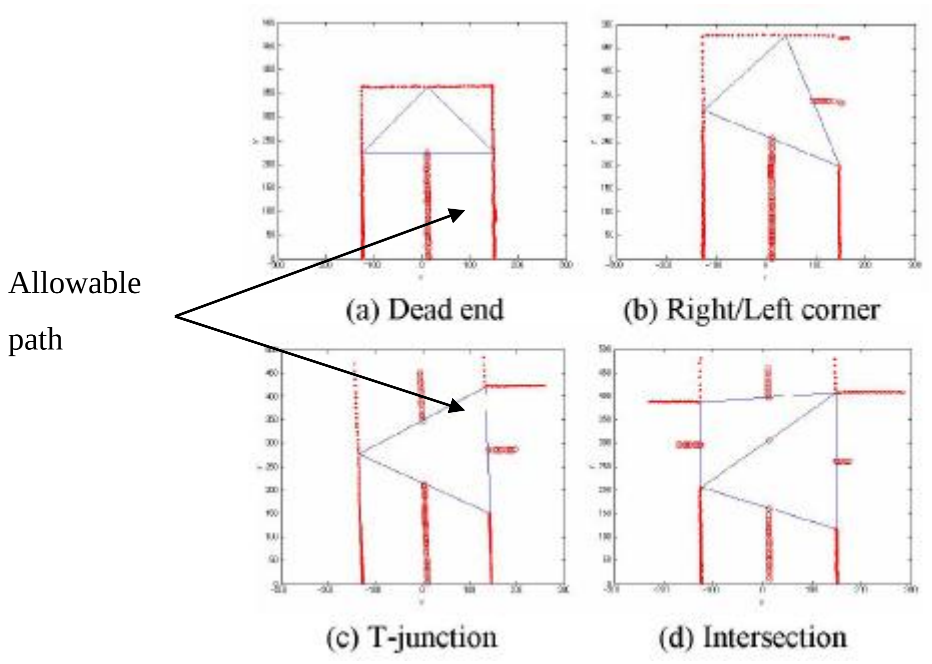


Figure 2.11: Landmark Triangle

Step 4 is the rules that need to be applied for structure recognition.

Dead-end: At least one allowable path exists.

Left/Right: At least two allowable paths exist.

T-junction: At least three allowable paths exist.

Intersection: At least four allowable paths exist.

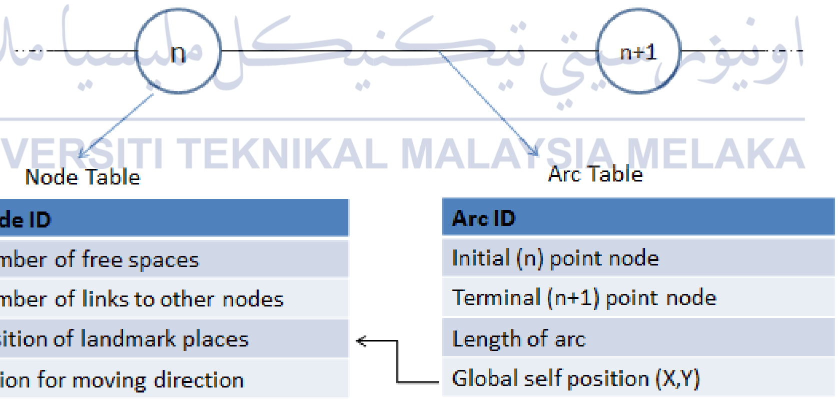


Figure 2.12: Topological map structure and labeling information

Figure 2.12 shows the important of labeling in generating the topological map for indoor navigation. Topological nodes are labeled with node ID, number of free spaces, number of links to other nodes, position of landmark place and action for moving direction. The topological arcs are labeled with arc ID, initial point node, terminal point node, length of arc and global self-position (X, Y).

j

i

	Node 1	Node 2	Node 3	Node 4	Node 5
Node 1	0	1	0	0	0
Node 2	1	0	1	0	0
Node 3	0	1	0	1	0
Node 4	0	0	1	0	1
Node 5	0	0	0	1	0

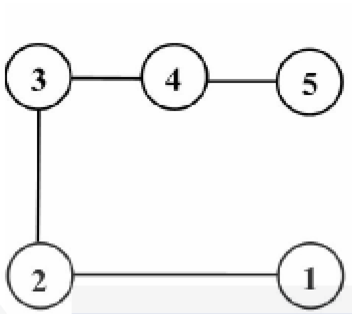


Figure 2.13: Undirected graph represented by adjacency matrix

Figure 2.13 shows the adjacency matrix method on the left hand side can used to solve the undirected graph obtained from topological map generation. The result of the adjacency matrix can used to produce the route as shown on the right hand side.

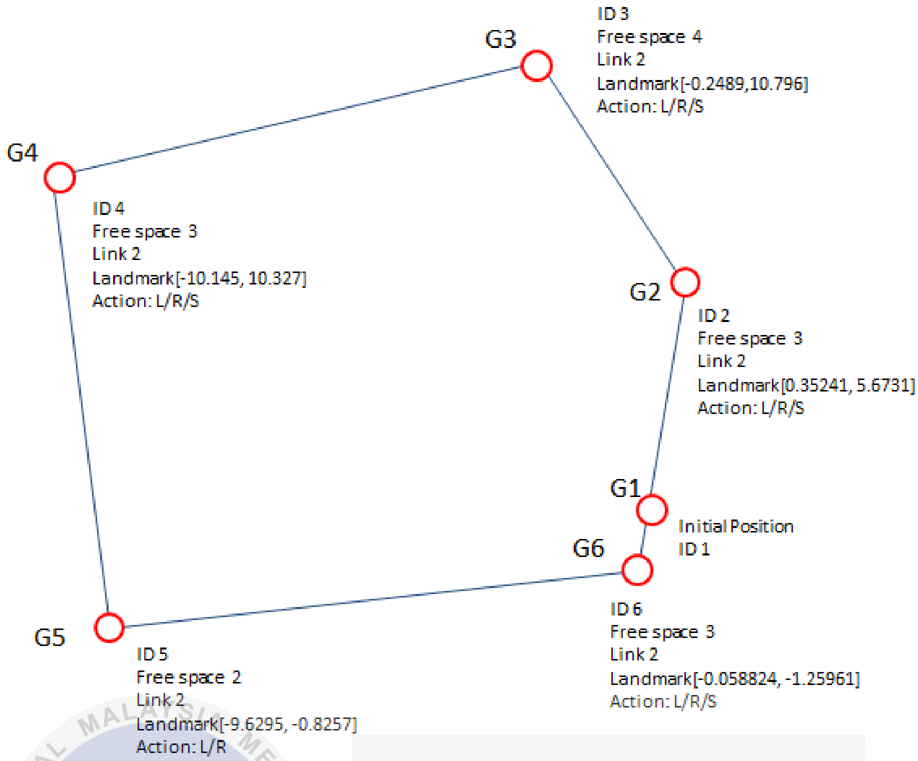


Figure 2.14: Topological map generation results

Figure 2.14 shows the effectiveness of the method in guiding the robot to complete the route. Point G1 represents the initial position, G2 represent a corridor T-junction, point G3 a corridor representation, G4 corridor T-junction, G5 corridor left corner and G6 a corridor T-junction.

2.5.3 Dijkstra Algorithm

N.A.Vlassis, N.M.Sgouros, G.Efthivoulidis and G.Papakonstantinou proposed global path planning method for autonomous qualitative navigation for robot. This method uses qualitative map (Qmap) of environment as shown in Figure 2.15 below. Qmap is built by applying grid consists of rectangular cell. The black cell represents walls or other obstacles and the rest space are free regions. Each cell contains the 8 proximity sensor facing 45 degree outside. Each sensor shows the orientation directions, N, NE, E, SE, S, SW, W and NW [22].

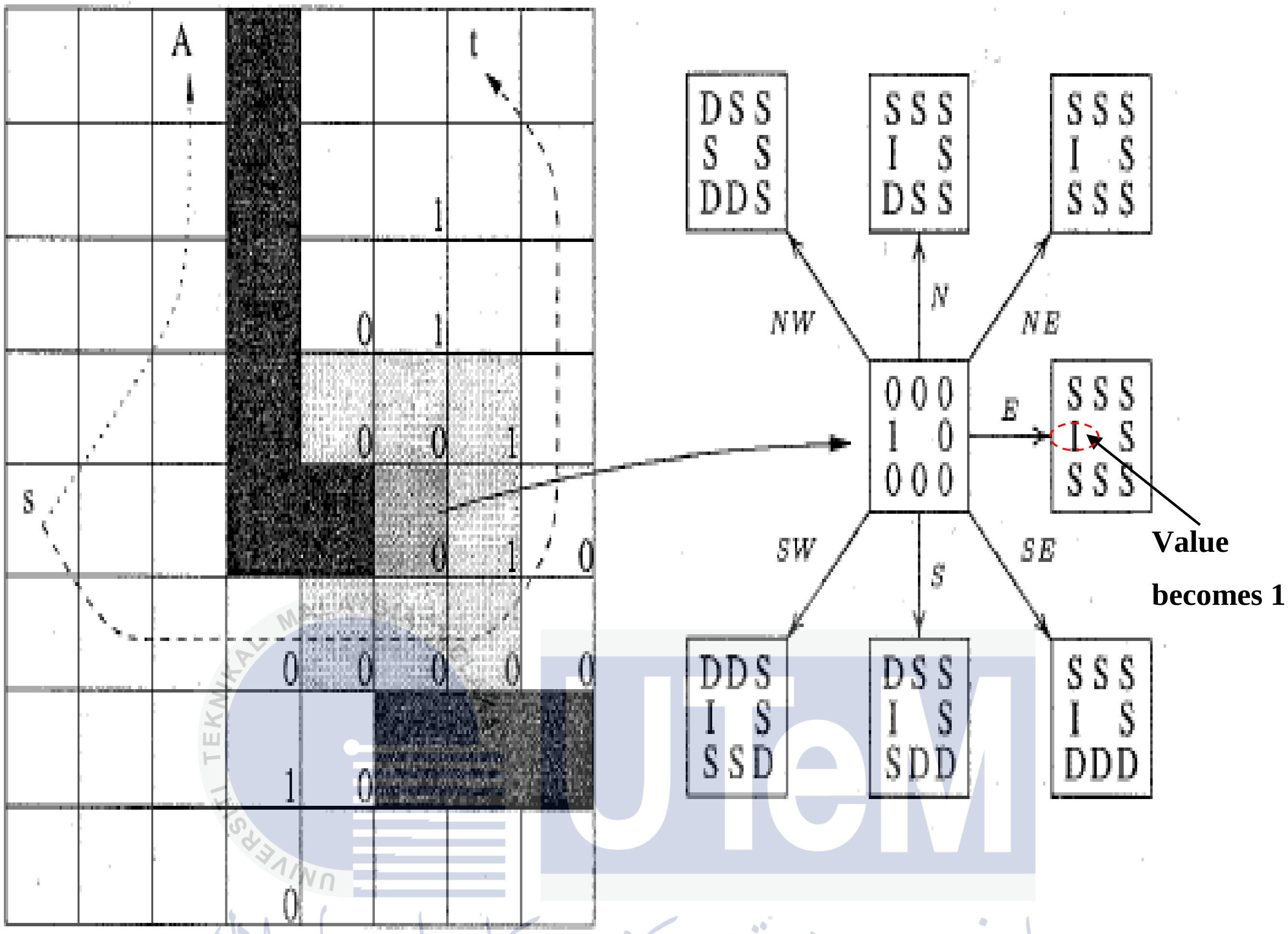


Figure 2.15: Qualitative map (Qmap)

The qualitative simulation starts when the robot moves from one cell to nearest cell. The movement of robot will lead to the changes of the sensor readings. The values of sensor show value=I (increased), value = D (decreased) or S (remain stable). The value of sensor decreases when the robot is near to obstacles. The value of the cell of Qmap is increased when the robot moves east while the other sensor does not face the robot will maintain the value (S).

The algorithm also uses the Dijkstra's algorithm. Two Sets S and B is used, set S holds the starting point and planned point to destination while B holds the successors of points.

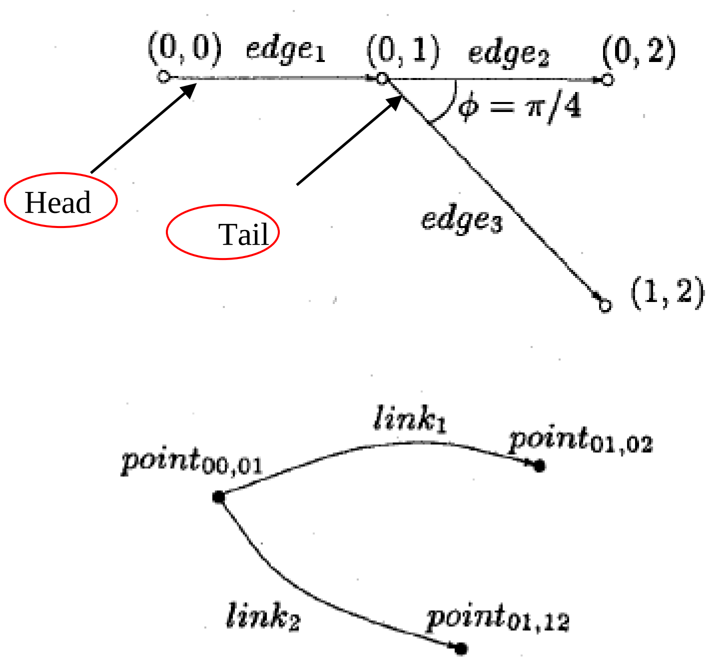


Figure 2.16: Nodes that link points

The two nodes that link a point are called head and tail. Node is (0, 0) and tail is (0, 1). The graph contains the point v and v' and there is a link (v→v'), then the point v' is called successor of point v. The link must contain the predecessor and successor points of v at the end of algorithm.

The length l of a link (v→ v') is computed as

$$L = |v| + |v'| + f.(1 - \cos \emptyset) + \omega.d$$

Link 1: L=1 + 1+ 0+ 1 =3 (2 straight transition an angle of 0° + off-wall distance equal to 1)

$$\text{Link 2: } 1 + \sqrt{2} + 1 - \cos\frac{\pi}{4} + 1 = 3/7 > 3$$

From the equation above, it is known that the trajectory through edge2 has a better than the edge3.

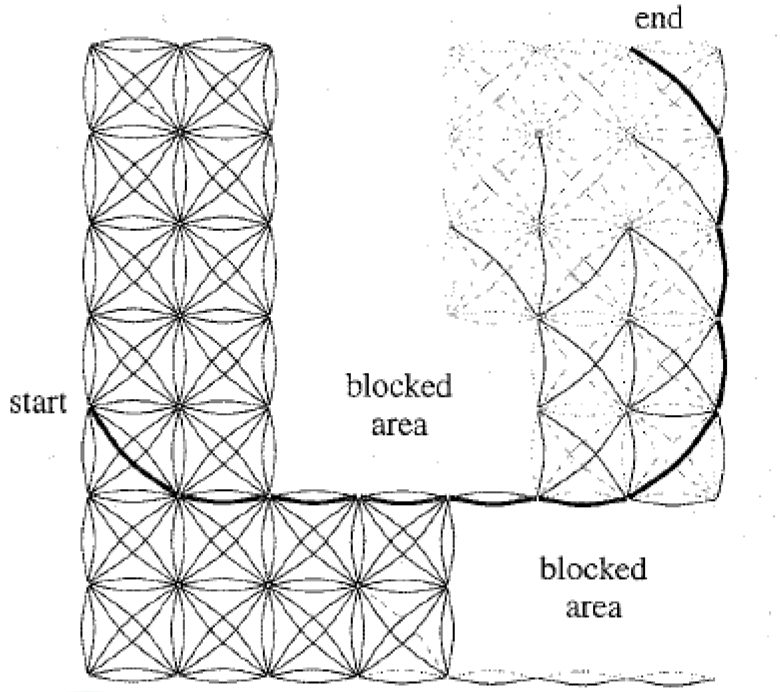


Figure 2.17: Trajectories from Start to End Point

Figure 2.17 shows the shortest distance from the starting to end point can be found by using the algorithm.

2.6 Summary of Literature Review

Considerations are taken for designing the navigation device with RFID network for visually impaired people after summarizing the literature review. The method and technologies chosen are based on the study and analysis from literature review. Decisions from literature review is made based on the weighing the pros and cons of the various method in order to come out better design.

RFID based navigation system for indoor navigation for the visually impaired people is chosen as the navigation device from this project. Visually impaired people will find the shortest path from his current location to destination with the aid this device. Besides, this device helps them when they lost their ways and recalculate the new route by using ARDUINO programming and digital compass module. RFID tags which are embedded into a footpath can be read by RFID reader.

Path algorithm that studied used to design an algorithm to find the shortest distance from the starting point to the end point. A* algorithm is chosen to develop the shortest path algorithm. Furthermore, knowledge gained from the navigation study can help to solve the problem when the user is out from the footpath and guide them back to their path normally.

GPS (Global Positioning System) does not work well at indoor navigation because the signals from the satellite are being blocked or attenuated by roofs, walls and other objects. Indoor positioning system such as RFID network can help in detecting people and objects, but it does not guide the visually impaired person for direction and orientation.

Since GPS cannot work well at indoor environment, accurate indoor maps are required for navigation purpose because locating people does not help the visually impaired people to travel independently. High cost is used to build an indoor map service provider. Therefore, a digital compass with high accuracy and low cost is an ideal solution.

CHAPTER 3

METHODOLOGY

3.1 Project Methodology

This chapter is about to present the method and procedure carried out to develop the hardware and software; and to evaluate the performance the designed prototype in order to achieve the objectives. The method that taken in order to complete the projects are shown below;

1. Study the literature review and information that related to the project, extract the useful information and propose the ideas of designing navigation device.
2. Search the suitable electronic components include sensors which are suitable for the project and study the datasheet.
3. Design and assembly the hardware
4. Assembly programming, download into the hardware and run debugging.
5. Hardware testing and compile, check, and edit the programming error.
6. Insulate the hardware with casing.
7. Experiments are carried out to test and evaluate the performance of developed navigation device.
8. Collection of simulation data and doing analysis
9. Final report writing

Figure 3.1 shows the flowchart of this project. Initially, literature reviews of journals are studied in order to understand and come out the idea of how to design the navigation device. Useful information is extracted to propose the idea of design to achieve the objectives. Next, search the components like sensors and RFID module that are suitable before design and assembly the blind navigation device. ARDUINO programming is compiled and downloaded to the hardware for debugging. Once all the components are tested properly, the hardware is insulated with casing for protection. Then, experiments are carried out to evaluate the performance of navigation device developed. Simulation data are collected and analysis is carried out to determine the validity and repeatability of the results. Lastly, final report is written for documentation of the project.

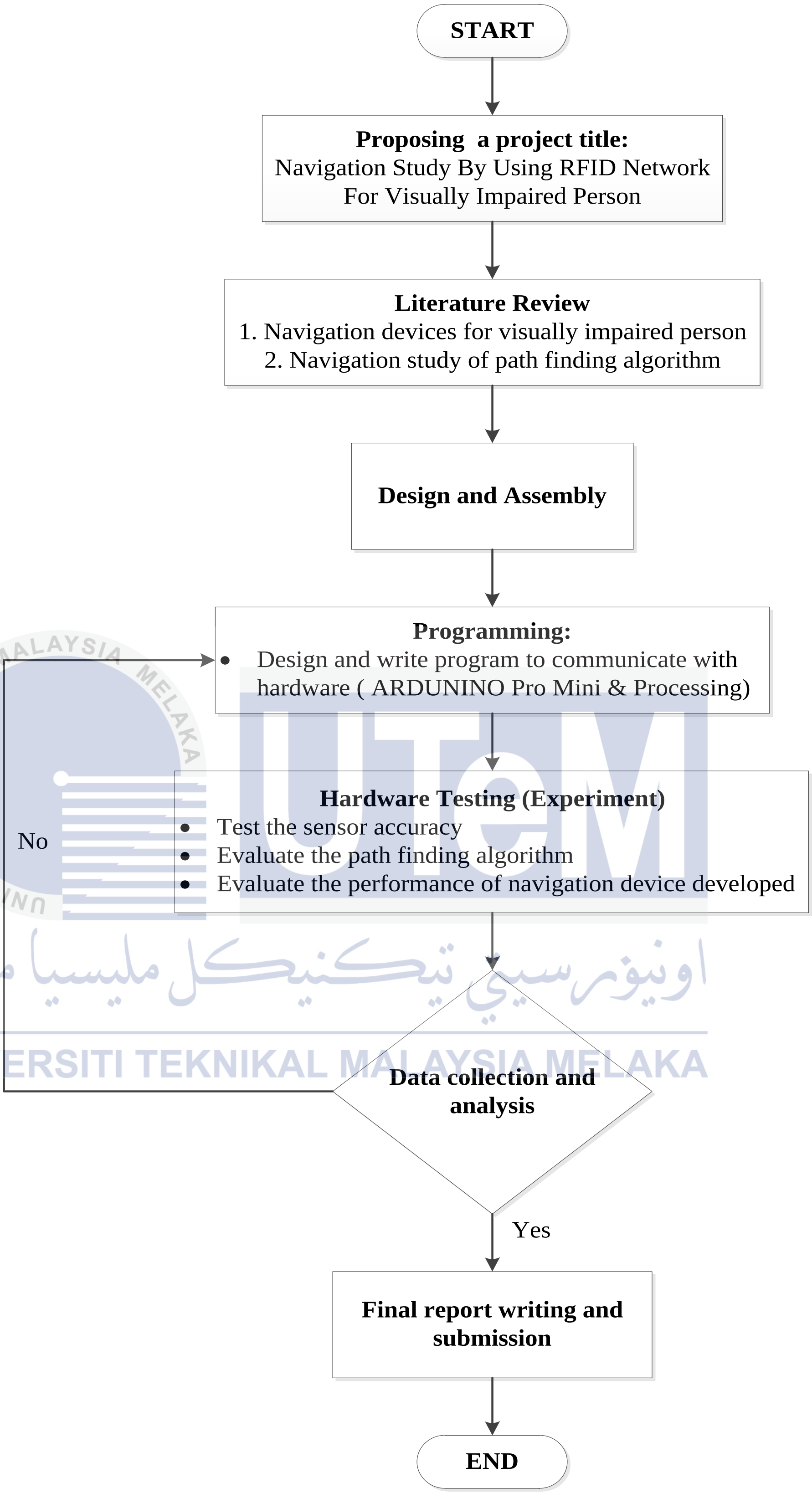


Figure 3.1: Flow chart of methodology

3.2 K-Chart

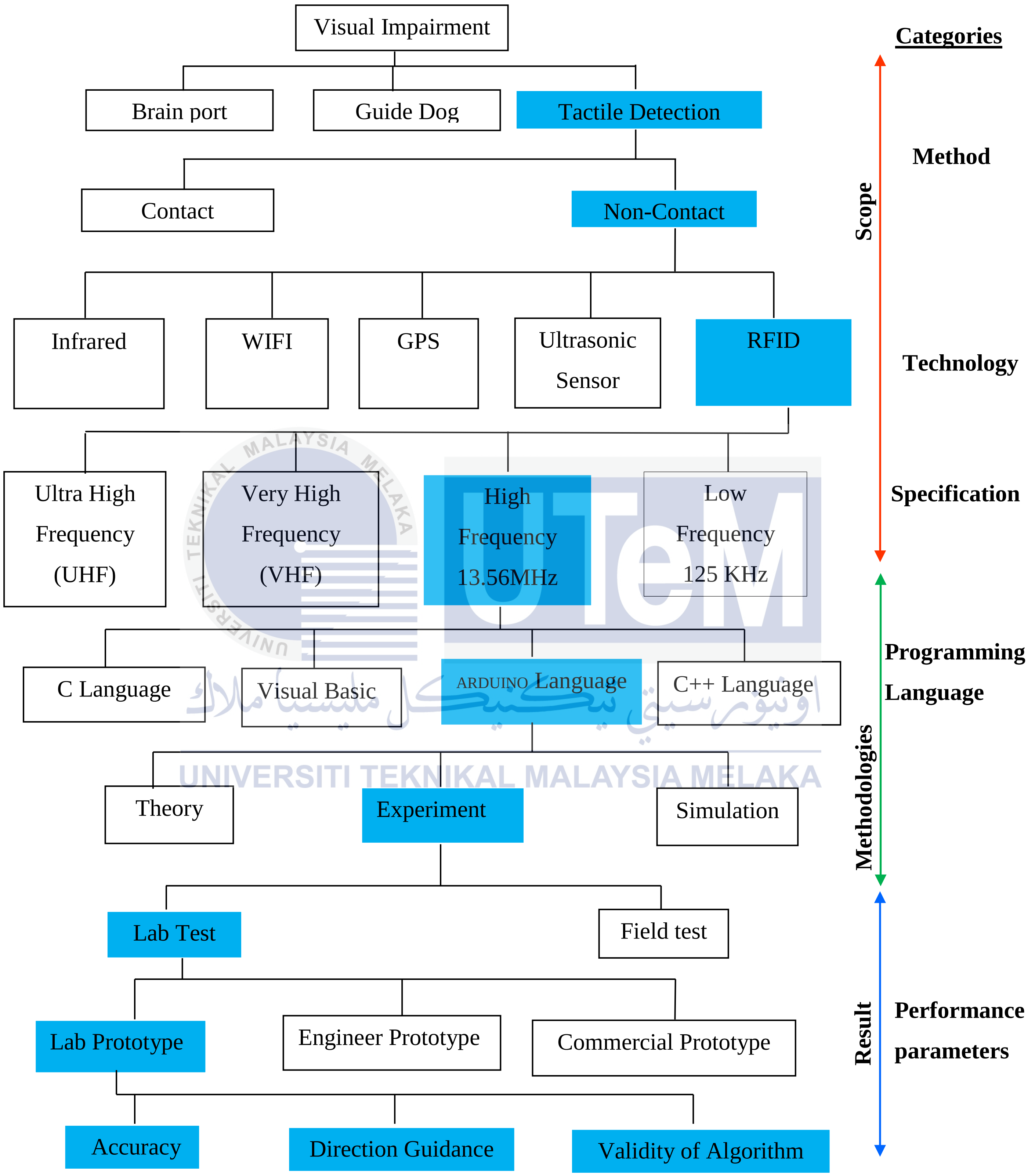


Figure 3.2: K-Chart for Navigation System for Blind

3.3 Design of the system

This section will discuss the design of the navigation device include Top View, Front View and Isometric View, and electronic components. There are two types of blind navigation device being assembled; both outlook structures are similar, only difference in what type of components is used. The second design will be used for this project in order to achieve the objectives.

3.3.1 Structural Design

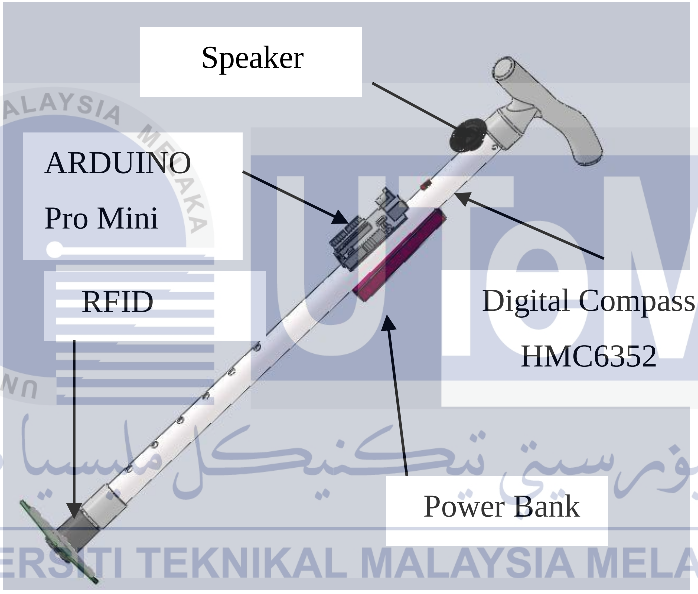


Figure 3.3: Navigation Device Full Drawing

The overall design is designed using Solid Works Premium 2012 which is shown in Figure 3.3. Walking stick is the main tool used to help visually impaired person to detect the obstacles during travelling. There are some electric components which are installed on the walking stick for navigation purpose. RFID reader is needed to read tags on the tactile paving since this RFID network technology is used for navigation purpose. Digital compass module HMC 6352 serial communicates with ARDUINO Uno main controller board to determine the direction where the user needs to direct to and when the users travel out of the direction of the path. Navigation information from the controller board will be transmitted to user via speaker or headset in the form of audio signal. Power bank is chosen as the main power supply because of its larger power capacity in mAH and it is portable and rechargeable.

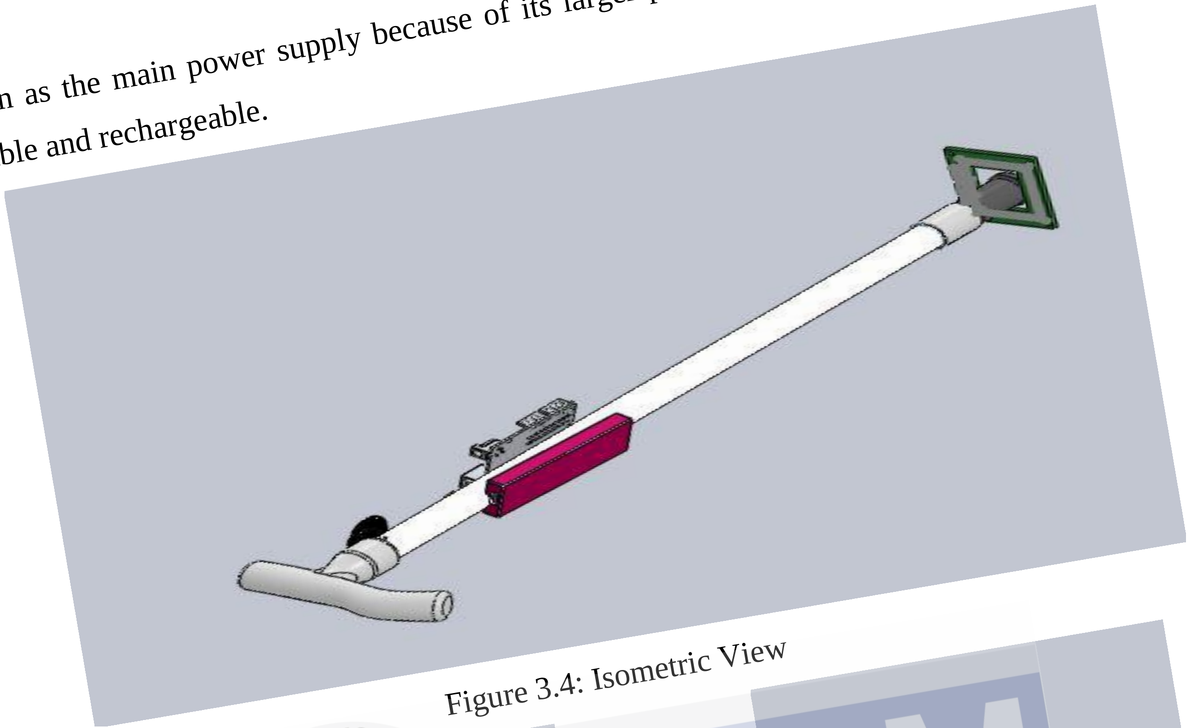


Figure 3.4: Isometric View

Figure 3.5: Top View

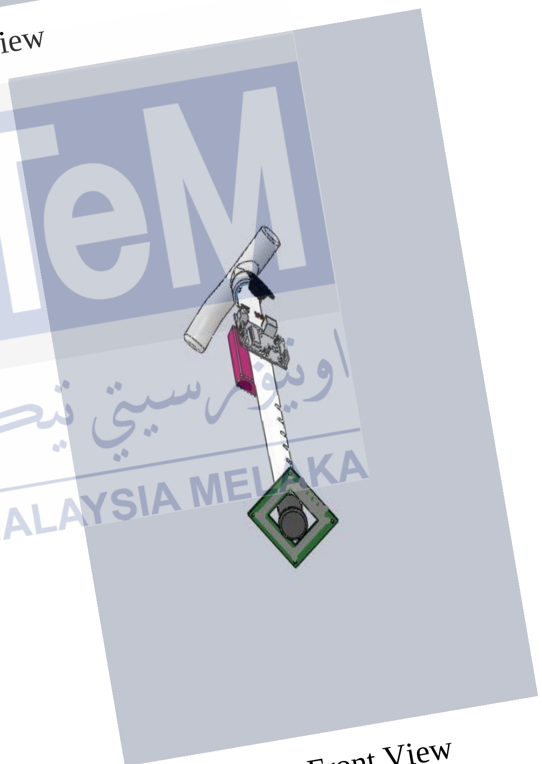


Figure 3.6: Front View

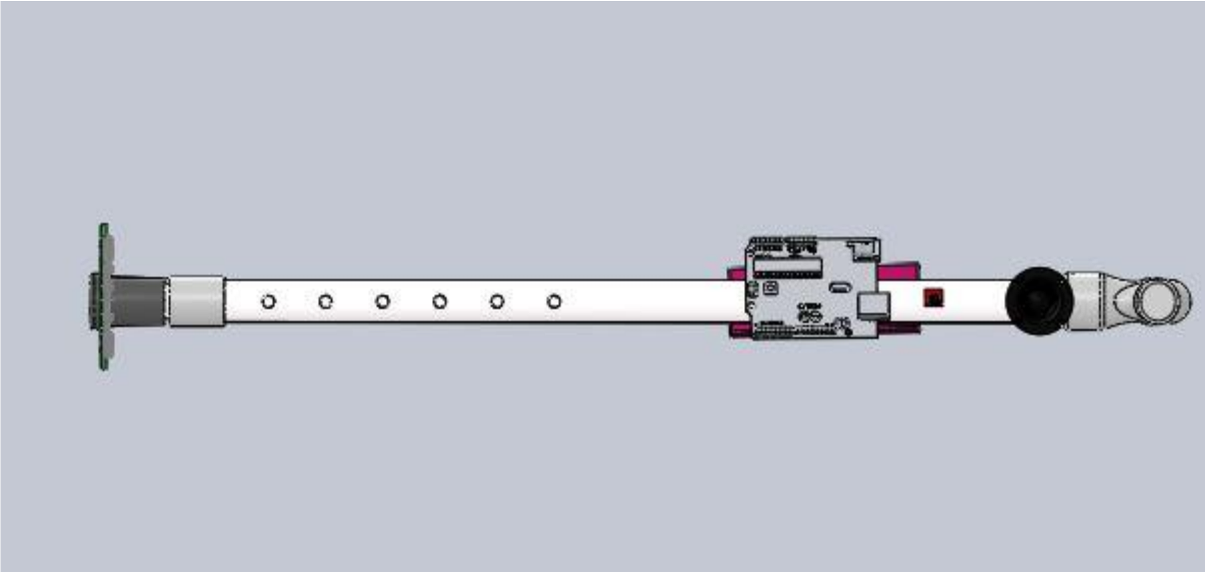


Figure 3.7: Left View

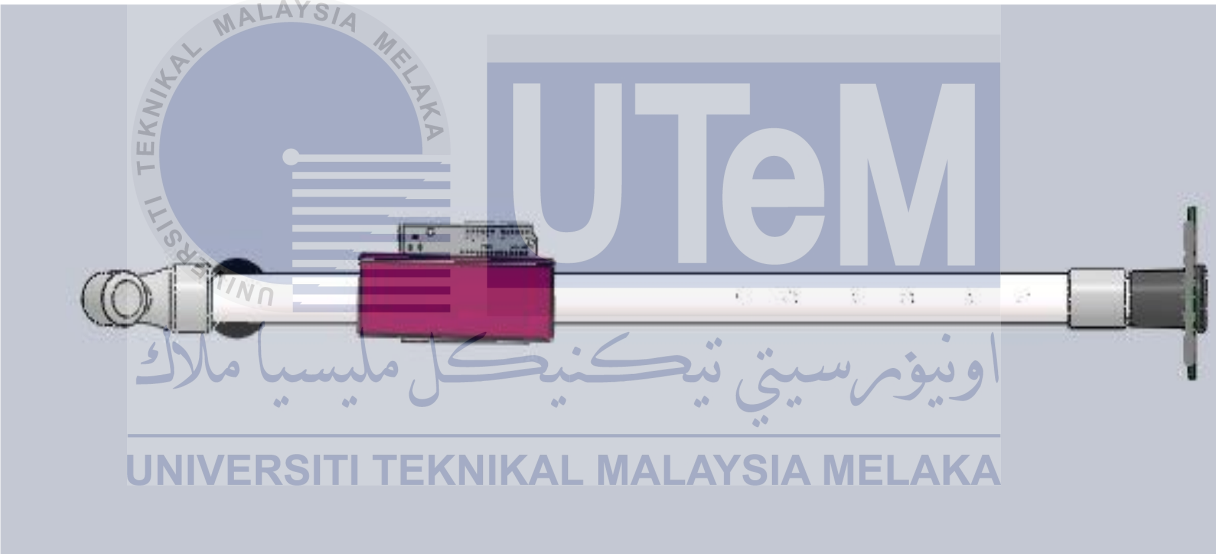
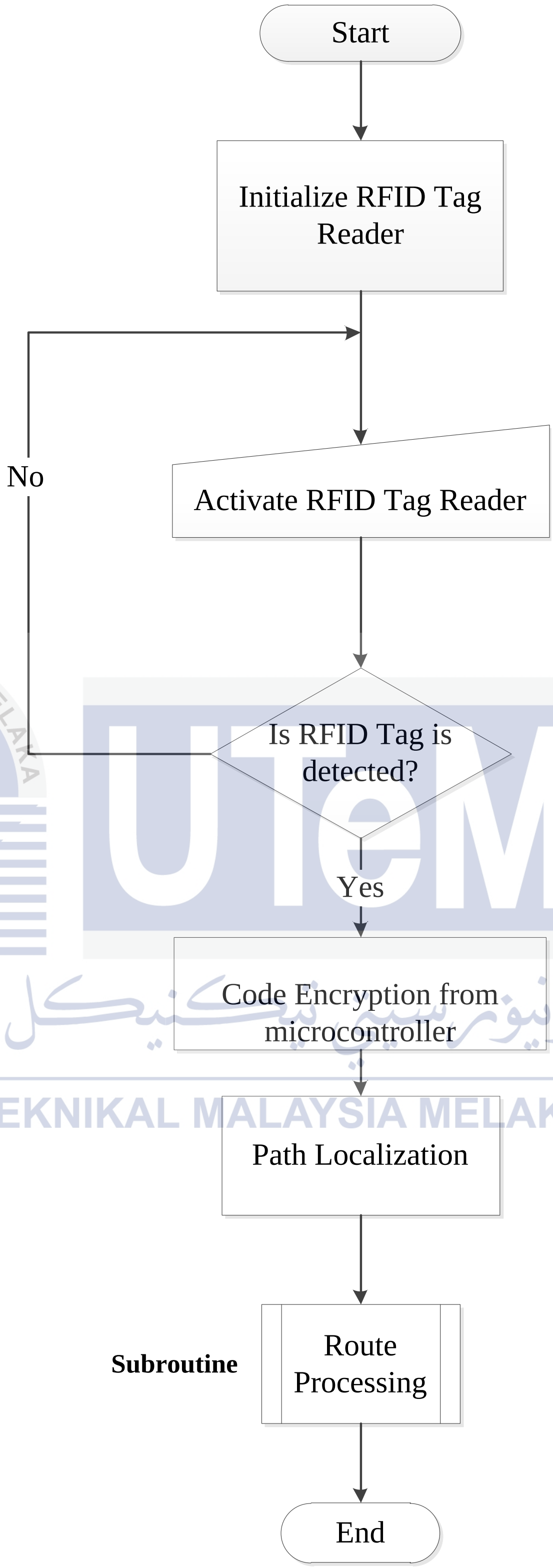


Figure 3.8: Right View

3.4 System Flow Chart



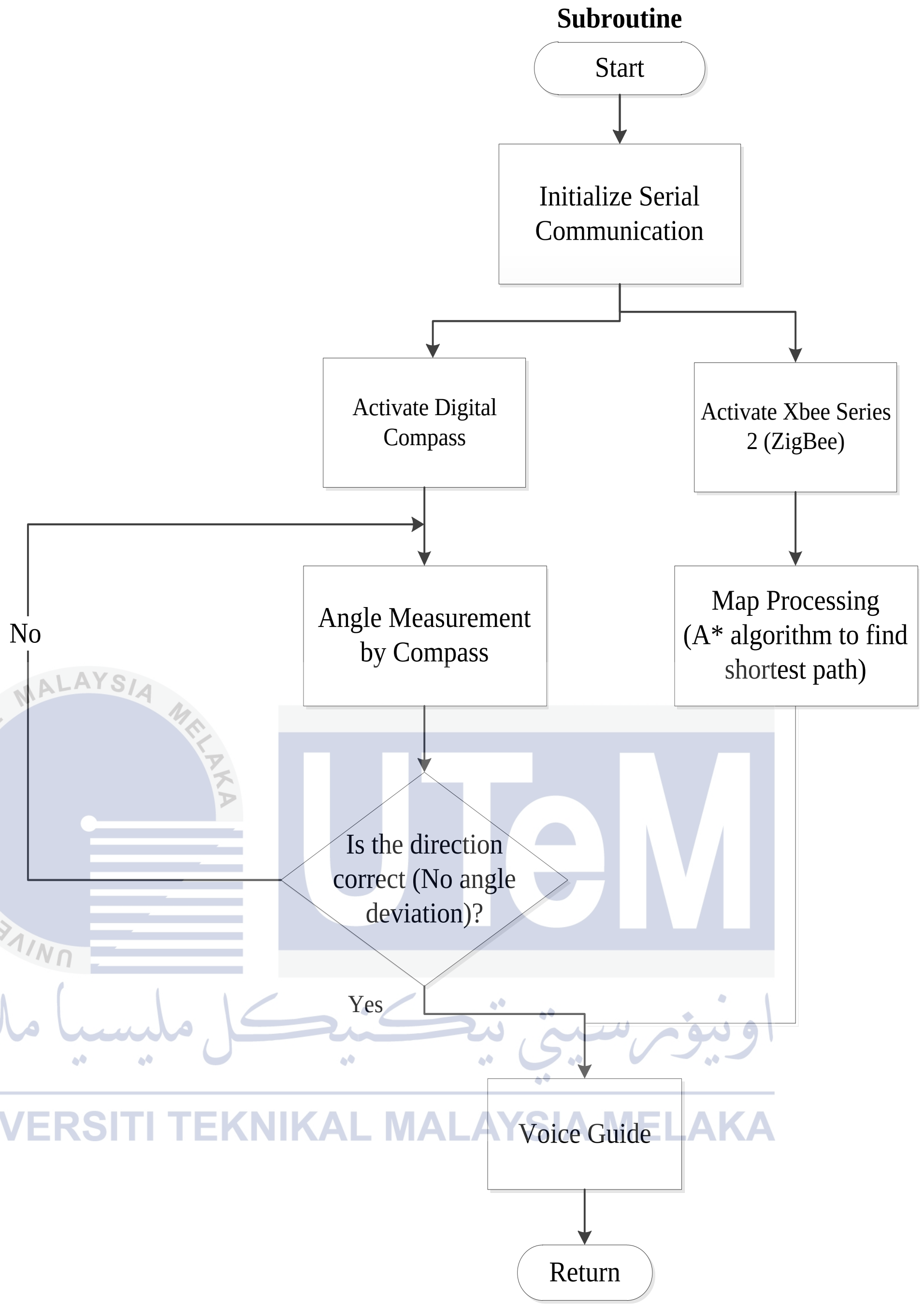


Figure 3.9: System Flow Chart

Figure 3.9 shows the flow chart of how the second navigation device works which is used throughout this project. Initially, the RFID tag reader is initialized. Once the tag reader has been activated, the user can key in the destination he or she wants to go using the modified keypad with Braille Code. The tag reader will determine the user's starting point and the device will start to guide the visually impaired people. If the RFID tags are successfully sensed, the ARDUINO microcontroller will carry out encryption of the tag identity code. If failed to do so, the device will go back to the tag reader activation process. After the identity encryption process, the device process to path localization will lead to user's desired destination guided by sound navigation. In the navigation subroutine, digital compass is activated by initialize the serial communication from the ARDUINO mainboard. The magneto sensor will measure the angle deviation and recalculate the correct heading direction. If there is no more angle deviation, the user can continue his or her travel through sound navigation and vice-versa. At the same time, the XBEE Series 2 (ZIGBEE) is activated as well. The decoded information of tags will be transferred wirelessly to the laptop for map processing. The A* algorithm written using Processing will determine the shortest path from the start node to goal node. Lastly, once the shortest path is determined, the voice commands are given to user through headset (e.g. Turn Right, Turn Left, Go Forward, and Warning). Warning commands usually given when the user heads to the other direction and deviate from the digital compass that always at "on" mode to measure the angle.

3.5 RFID Detection System for First Design

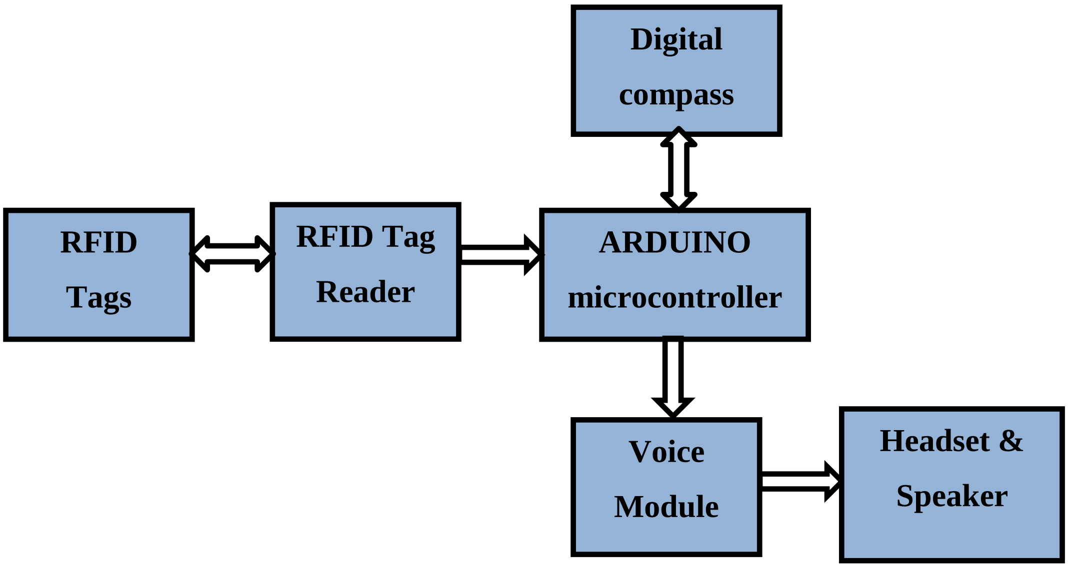


Figure 3.10: RFID detection system for First design

Figure 3.10 shows the block diagram of the RFID detection system for the first design of the blind navigation device. The ARDUINO Pro Mini microcontroller is the brain of the RFID detection system which responsible for information processing and decision making to produce the next action. Initially, RFID tag reader is activated to read the tags on the tactile paving in terms of bytes of data. Next, the microcontroller encrypts the data and converts into useful information. It is always the two ways communication between RFID tag reader and tags. At the same time, the digital compass that connected to the microcontroller is always activated to obtain the angle when the user is walking. The other function of digital compass is to compare the angle whether it is inside the range that is preset in the programming code to ensure the user's orientation is correct. With the combination of tags information and correct angle, the microcontroller will send the voice commands through the voice module. Voice module contains various commands such as move forward, turn right, turn left and warning. Finally, the user can receive the clear commands to travel the path through headset or speaker conveniently.

3.6 Electronic Design

The components used in the first navigation device are:

a) ARDUINO Pro Mini microcontroller
b) MIFARE RFID module (RFID reader and tags)
c) Digital Compass Module
d) Voice Module WTV020-SD
e) Earphone
f) Power Bank

3.6.1 ARDUINO Pro Mini

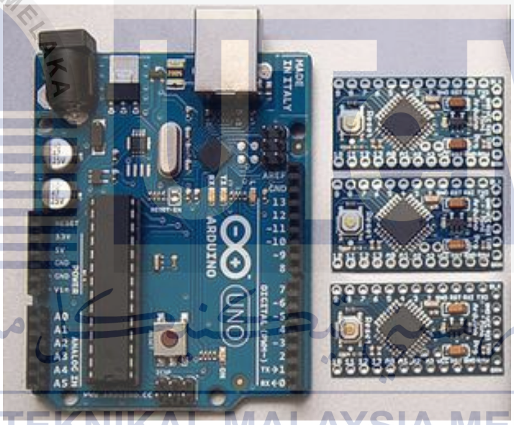


Figure 3.11: ARDUINO Pro Mini

ARDUINO developers come out varies kind of single board microcontroller which vary from mid-range to high-range specifications. ARDUINO Pro Mini is one of the single-board microcontrollers that are designed for advanced users who required low cost, low power consumption, light weight, small size and flexibility. 3 ARDUINO Pro Mini is equivalent to one ARDUINO Uno and yet the features for both are the same; USB downloader is excluded from the board and no pin socket is soldered. Specification of ARDUINO Pro Mini is attached in Appendix C.

3.6.2 MIFARE RFID Reader Module

Figure 3.12: MIFARE RFID Reader Module

Radio-frequency identification detection (RFID) is a wireless technology that uses of radio frequency electromagnetic fields for the purpose of data transmission and identity detection. High Frequency (13.56MHz) is used for this project. Specification of MIFARE RFID Reader Module is attached in Appendix D.

3.6.3MIFARE High Frequency RFID Tag

Figure 3.13: MIFARE High Frequency RFID Tag

RFID tags can be categorized into different types such as passive, active or battery-assisted passive. Tag is read only and it has the unique identification code that is used to access into the database. The coil inside the tag is activated by converting the magnetic flux produced by RFID reader. Reading range can be reached more than 10km.

3.6.4 Digital Compass

Figure 3.14: Digital Compass HMC6352

Digital compass provides the straight-forward degree resolution compass heading. The reason of using digital compass is because it can work well with ARDUINO microcontroller and able to provide the digital reading instead of analog reading. The digital compass is able to produce high accuracy degree reading. Specification of digital compass (HMC6352) is attached in Appendix E.

3.6.5 WTV020-SD Voice Module

Figure 3.15: WTV020-SD Voice Module

WTV020-SD voice module is a voice rewritable voice storage device with higher memory capacity and it cans pluggable type. The voice can be changed by SD card reader and PC. SD card should be FAT format. Sampling rate supported from 6 KHz~32 KHz and 36 KHz for ad4 voice format. 6 KHz~16 KHz for WAV voice format. Specification of WTV020-SD module is attached in Appendix F.

3.6.6 Tactile Paving

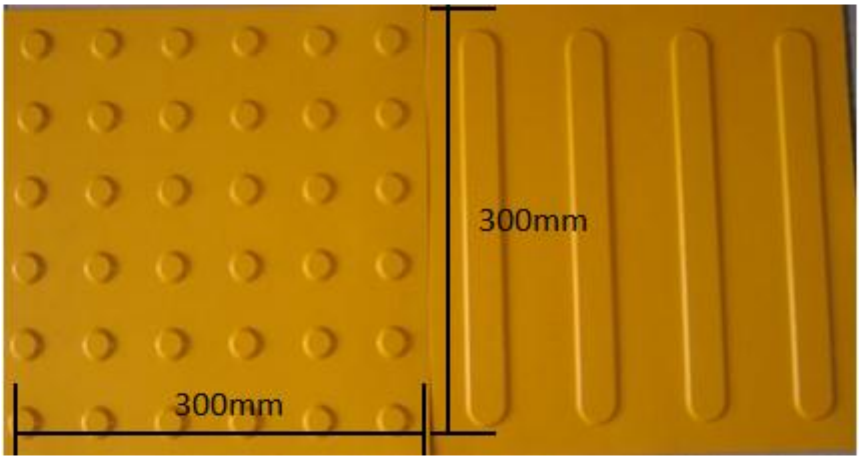


Figure 3.16: Tactile Paving

Tactile paving is texture ground surface designed for visually impaired person. It is kind of assistive tool to guide to travel from point to point. The size of tactile normally used is 300mmx300mmx6mm.

3.6.7 First Generation of Blind Navigation Device

Figure 3.17: First Generation of Blind Navigation Device

3.6.8 First Design Schematic Diagram

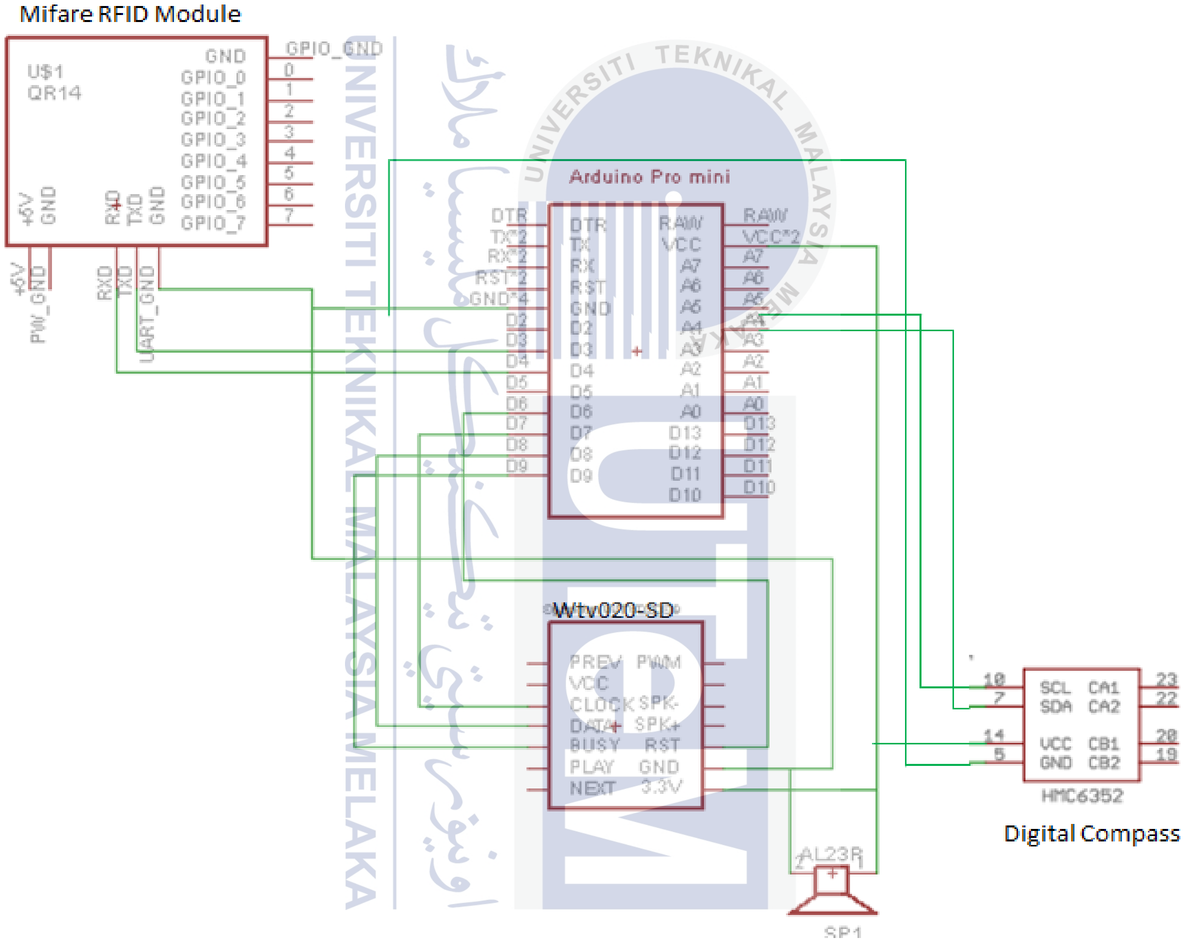


Figure 3.18: First Design Schematic Diagram

3.6.9 First Design Circuit

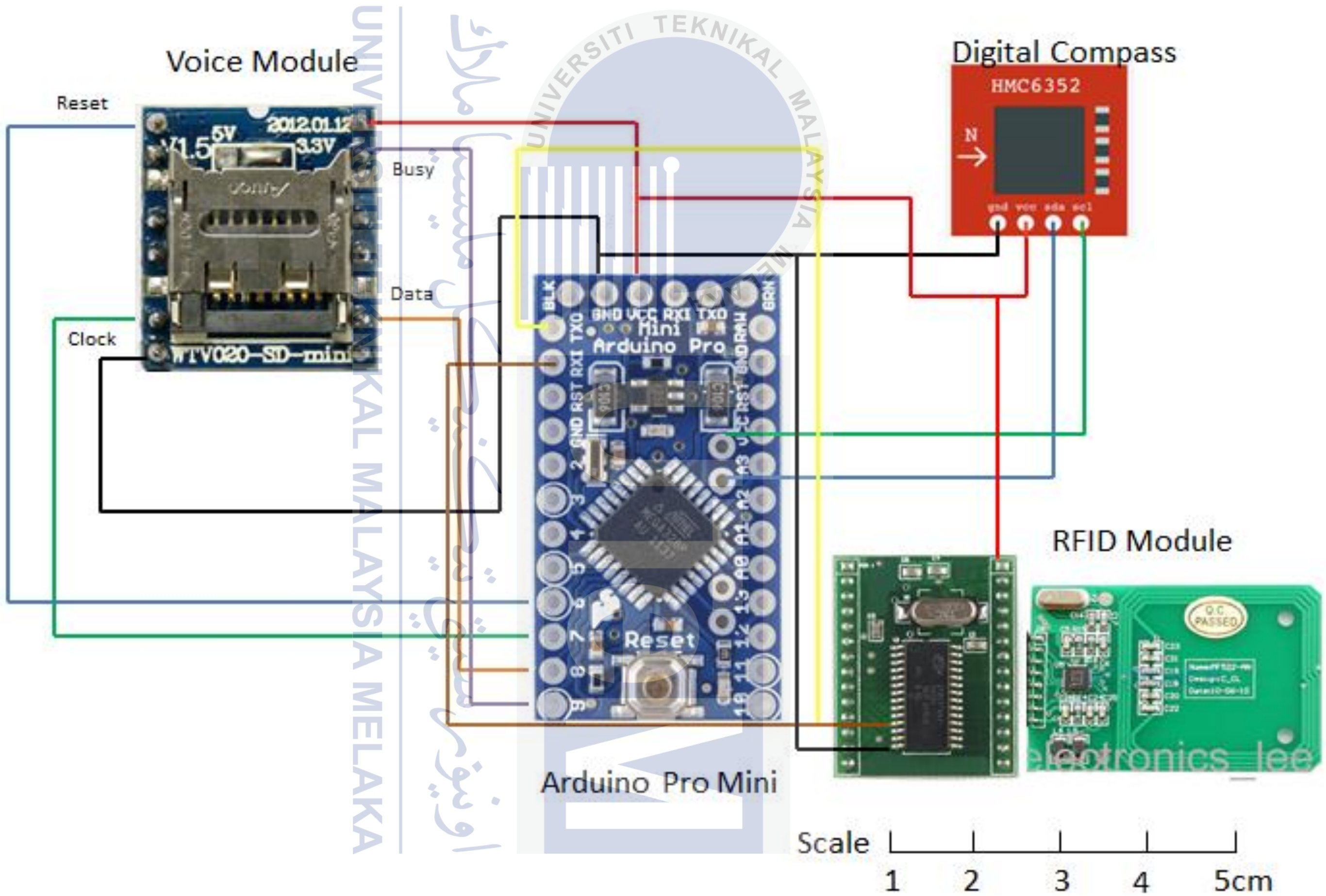


Figure 3.19: First Design Circuit

3.7 RFID Detection System for Second design

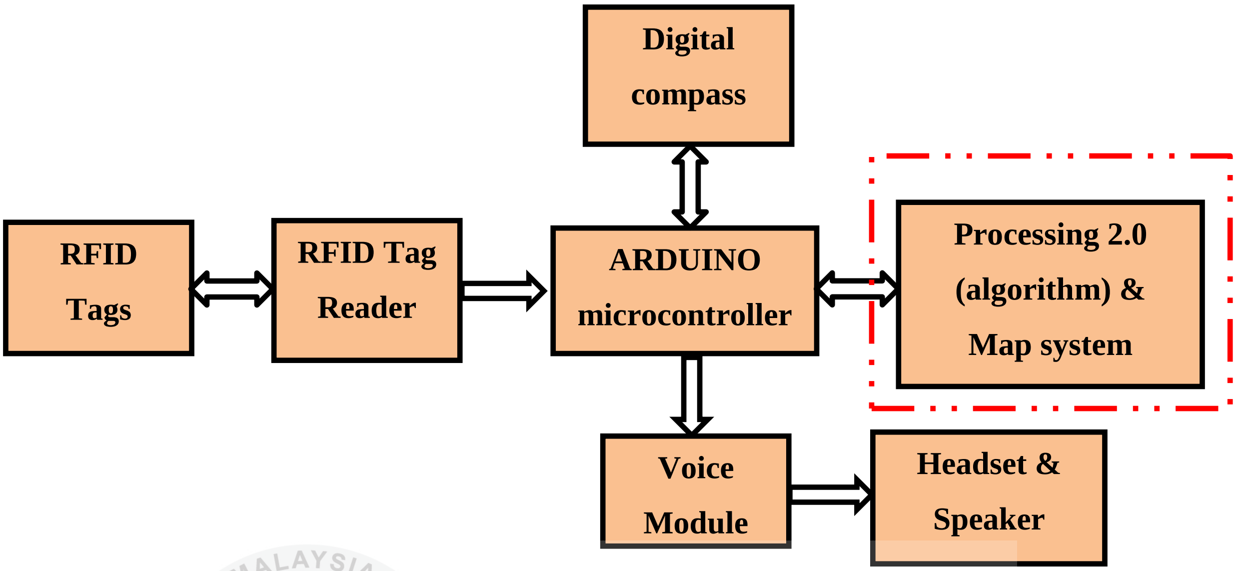


Figure 3.20: RFID detection system for Second design

Figure 3.20 shows the block diagram of the RFID detection system for the second design of the blind navigation device. Basically the functionality of both the RFID detection system is nearly the same. However, there is an improvement from the previous functionality. It can be said that the only difference between both the RFID detection systems is the second design synchronized with A* algorithm through XBEE wireless communication. This blind navigation device is connected to the laptop through wireless communication whereby one ZIGBEE is connected to electronic cane while the other ZIGBEE is connected to laptop. Both the ZIGBEE is connected together once they are powered on. Meanwhile, the map system created using Processing 2.0 software is synchronized. Current position will be shown on the map once the tag is detected. Furthermore, the A* algorithm will be shown if the initial location and final destination is known. With the information from the map system and shortest path, the visually impaired people can travel independently and safe guided with the voice commands through the headset.

The components used in the second navigation device are:

a) ARDUINO Pro Mini microcontroller
b) MIFARE RFID module (RFID reader and tags)
c) Digital Compass Module
d) Voice Module WTV020-SD
e) BLUETOOTH module
f) BLUETOOTH headset
g) XBEE Series 2(ZIGBEE)
h) Modified keypad with Braille Code
i) Power Bank

3.7.1 XBEE Series 2 (ZIGBEE)

XBEE Series 2 has enhanced its power output and data protocol. Series 2 can create complex networks based on the XBEE ZIGBEE Mesh firmware. This module provides reliable communication between microcontrollers and computer system.

Figure 3.21: XBEE Series 2 (ZIGBEE)

3.7.2 Numeric 4x4 Matric Keypad

This 16 Key 4x4 membrane Alpha Numeric Keypads looks like standard telephone type keypad which is suitable for microcontroller projects. It works same like the common keypad and requires user to give command to the system.

Figure 3.22: Numeric 4x4 Matric Keypad

3.7.3 Braille Code

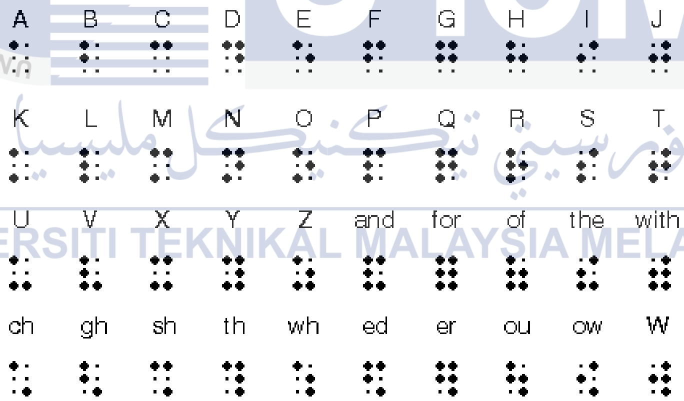


Figure 3.23: Braille Code

Figure 3.23 shows the Braille Code. It is a system designed for visually impaired person to read and write through touch. Each cell consists of 6 raised dots of pattern configuration of 3x2. Each cell represents the alphabet, punctuation marks, and letters.

3.7.4 Modified Keypad with Braille Code

Figure 3.24: Modified Keypad with Braille Code

Figure 3.24 shows the modified keypad with Braille Code. This Braille writing system is attached and connected on the top the 4x4 numeric keypad. It is special modified for key in the destination with the combination alphabet “T, A, N, D, S, U, R, M, P, L, F, O, K, E, L, #”.

3.7.5 Second Generation of Blind Navigation Device

Figure 3.25: Second Generation of Blind Navigation Device

3.7.6 Second Design Schematic Diagram

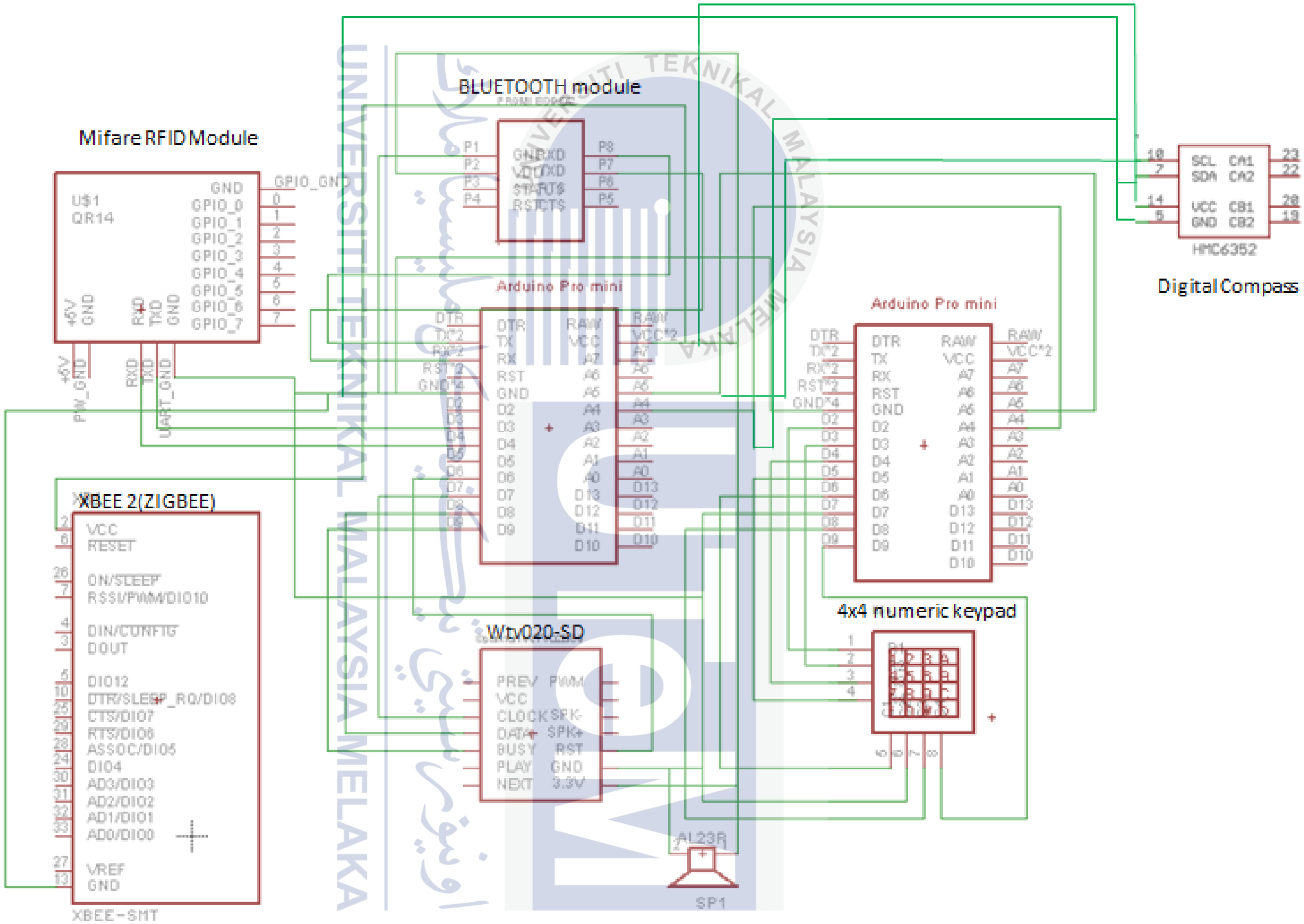


Figure 3.26: Second Design Schematic Diagram

3.7.7 Second Design Circuit

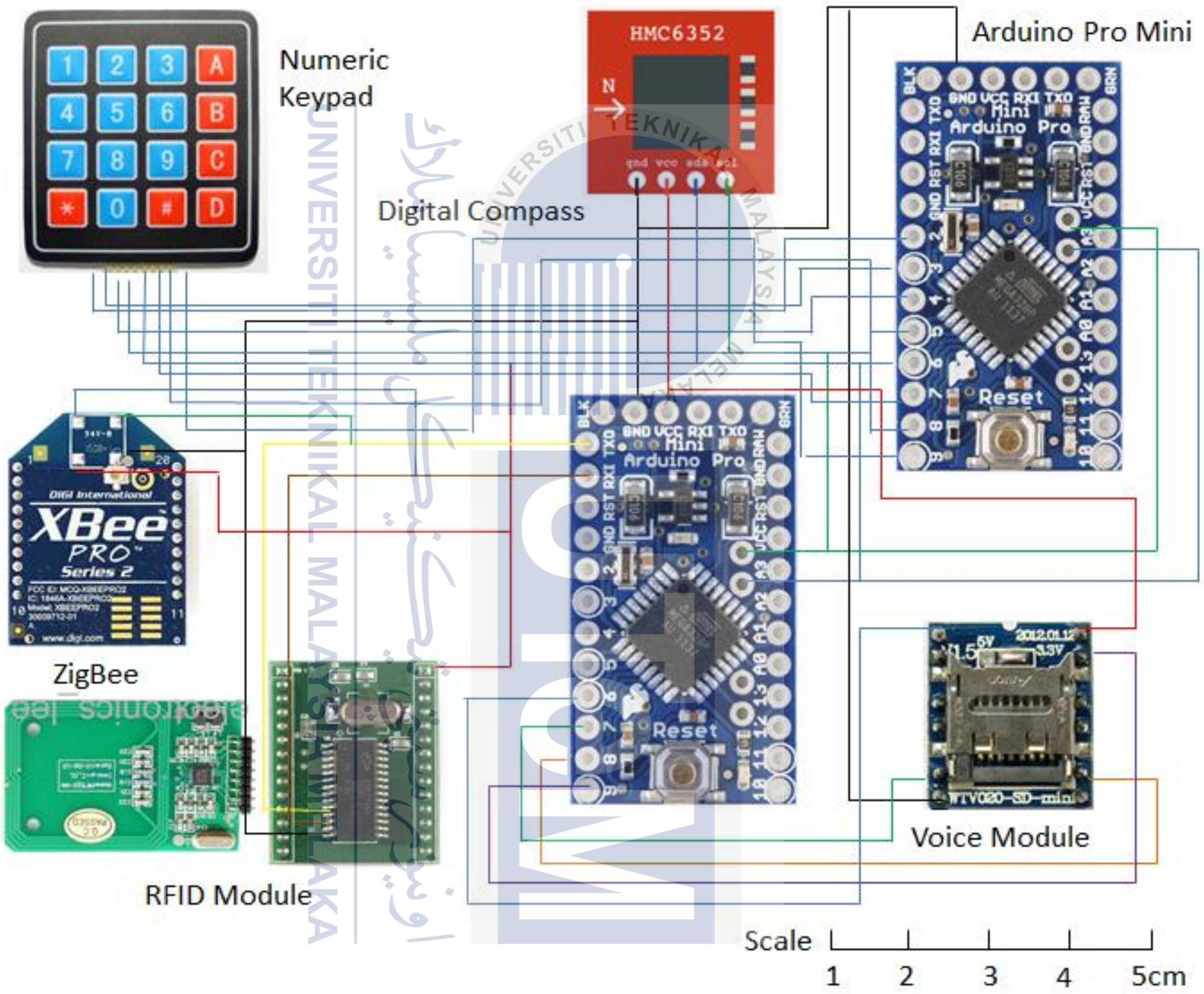


Figure 3.27: Second Design Circuit

3.7.8 Connection of XBEE to laptop

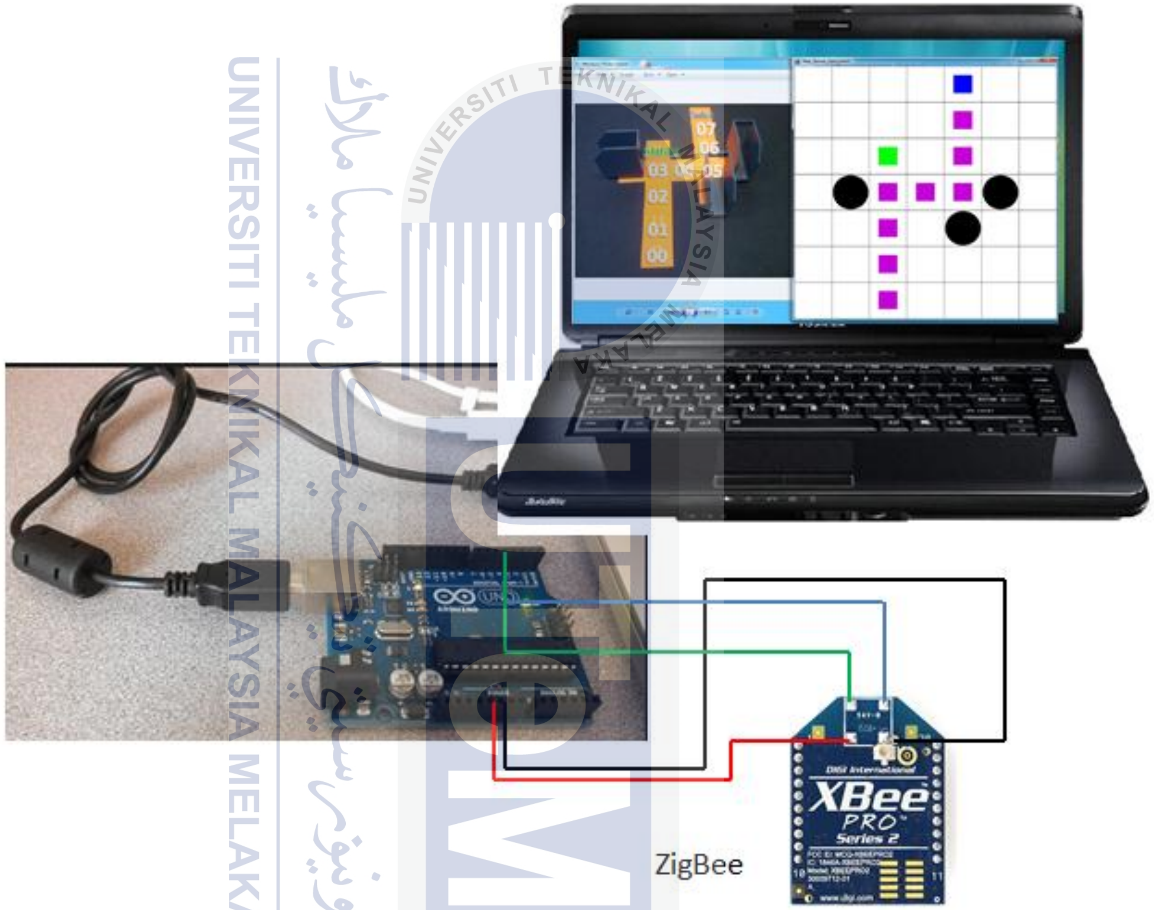


Figure 3.28: Connection XBEE to laptop

3.8 System Overview

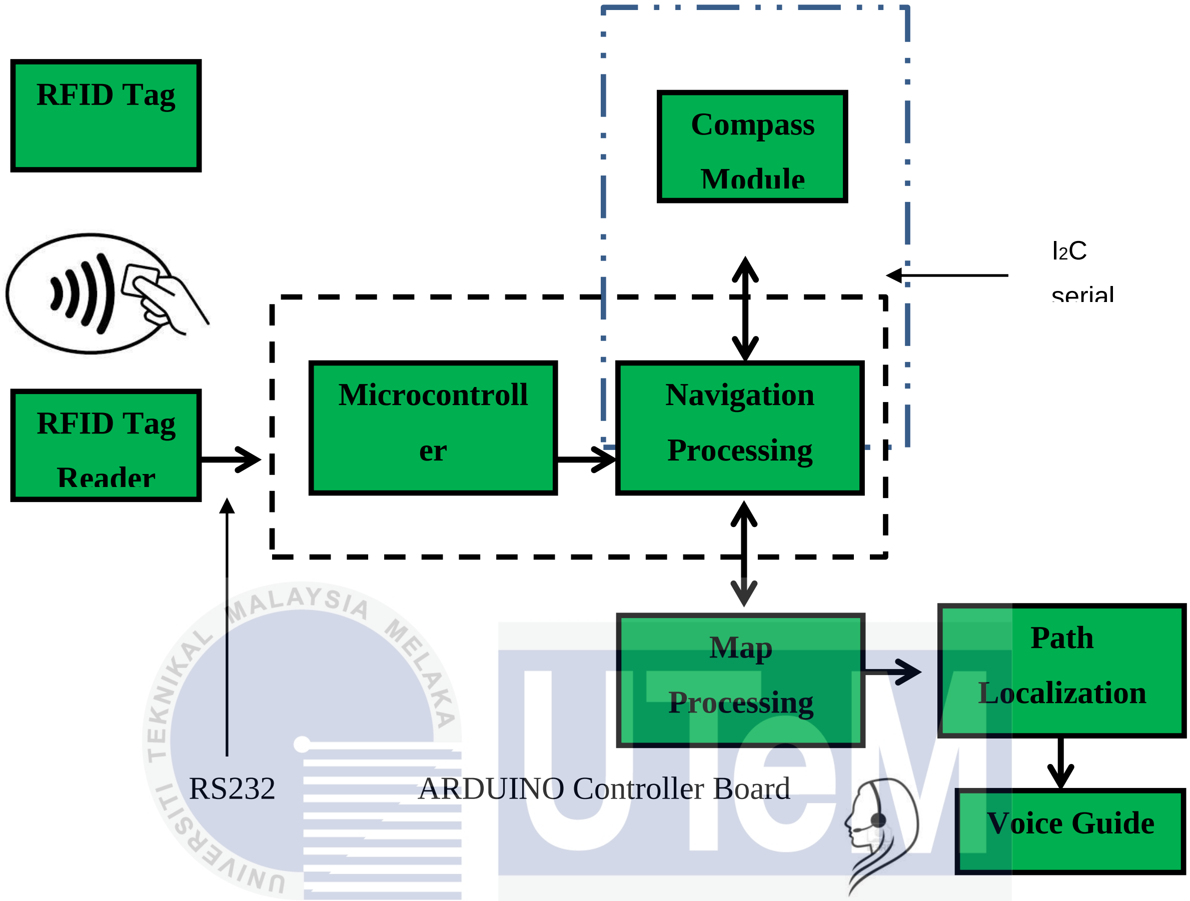


Figure 3.29: System overview of navigation device

Tactile paving is the travelling path for the visually impaired people. There are RFID tags installed on the tactile paving. RFID tag reader is detached on the walking stick to read the tags. Initially, user set the desired destination. RFID tag reader is activated to read the tags. Each RFID tag has its own identification code and the code is transferred to ARDUINO microcontroller through RS232 for identification code encryption. Next, the controller system proceeds to the navigation processing. The microcontroller receives map processing information about the shortest path from start node to goal node from laptop via ZIGBEE wirelessly. If the user travels out from the path, the navigation system will determine the direction heading and gives alert to user. Once the direction is correct, user can continue their travel by the aid of sound navigation with the shortest path.

3.9 Experiment

There are three experiments which are carried out throughout the whole project;

1. Experiment 1: Digital Compass Accuracy Test
2. Experiment 2: Mobile Robot Navigation to test Map System with A* Algorithm
3. Experiment 3: Lab Test of Blind Navigation Device

Research Question (Experiment 1)

1. Is the compass always pointing the north?
2. Is the visually impaired person lost direction easily when turn right or left?

Hypothesis:

1. The bearing angle of the compass might be affected and thus the accuracy is lower if placed near the environment under strong magnetic field influence.
2. Visually impaired person tends to lost direction if they turn too much and move far away from the RFID tags detection.
3. Visually impaired person is not able to follow direction if the swinging of the walking stick is larger.

3.9.1 Experiment 1: Digital Compass Accuracy Test

Objective**:** To determine the accuracy of the digital compass

Materials: ARDUINO Uno Microcontroller, Digital Compass HMC6352, real compass or IPHONE compass app as compass reference

In order to determine the accuracy of the digital compass that will be later implemented on the blind navigation device, the digital compass was connected to ARDUINO Pro Mini to run on a test. The digital compass module was turn on and the observation was recorded to check whether the value the serial monitor is 0° or 360°. The data was then collected to indicate the performance of the digital compass module. The raw data can be analyzed using the statistical method.

The procedure was summarized as below;

Procedure:

1. The digital compass module was connected to the ARDUINO microcontroller.
2. The digital compass module was taped down and the IPHONE compass app or real compass as the reference system is made as shown in Figure 3.30.
3. The digital compass was checked several times to ensure it fixed and standstill on its position.
4. The observation of the digital compass module from the serial monitor of ARDUINO software was noted.
5. Measurement was repeated for about 20 times to check the accuracy of digital compass whether it is always pointing to the NORTH.
6. The raw data obtained from the repeatability test can be analyzed by calculating the average (mean), standard deviation, relative error and percent relative error.

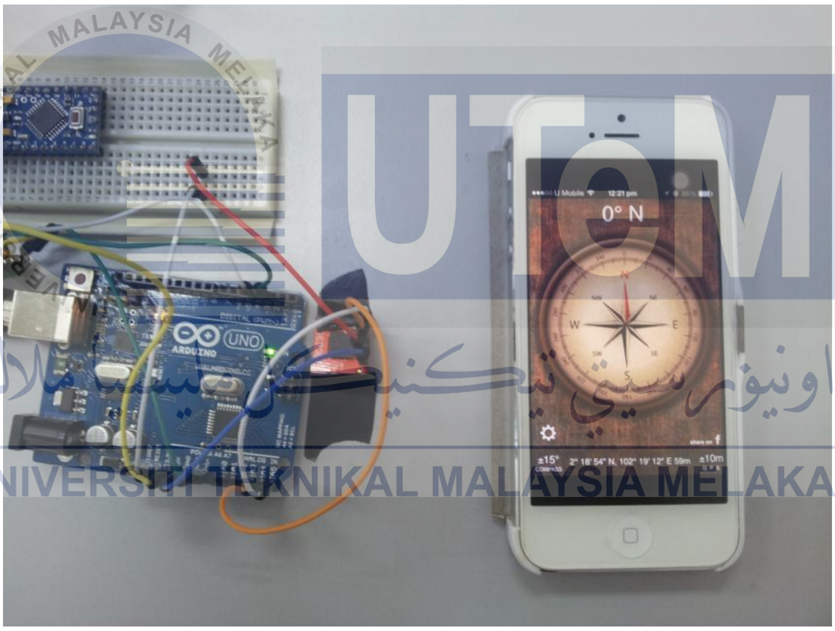


Figure 3.30: Digital compass and the reference compass

3.9.2 Experiment 1 Parameter

The raw data which is the orientation angle is measured by using the reference angle from the IPHONE Compass Apps or real compass. The raw data from the digital compass module can be obtained the serial monitor of the ARDUINO software.

Research Question (Experiment 2)

1. Is the A* algorithm is applicable for finding the shortest path on the map system?
2. Is there any difference where the A* algorithm applied on the navigation for mobile robot and blind navigation device?

Hypothesis:

1. A* algorithm is suitable for finding the shortest path from start node to goal node on the map system.
2. There might be difference in time when the mobile robot and blind people reach the same destination. Visually impaired person may take longer time to reach destination because need time to adapt to darkness condition and digest the information.

3.9.3 Experiment 2: Mobile robot navigation to test Map system with A* Algorithm

Objective: To determine the validity of the A* algorithm used in the map system.

Materials: Mobile robot, tactile paving, RFID tags, three obstacles, two cylindrical-shaped border marker.

In order to determine the validity of the A* algorithm used in the map system, a mobile robot with all the RFID module was set up. The mobile motor was required to travel on the tactile paving. The purpose of using the border marker is to check the number of times the mobile robot hitting on it. The time for mobile robot to reach the destination was recorded. The data was then collected to indicate the performance of the A* algorithm. The raw data can be analyzed using the statistical method.

The procedure was summarized as below;

Procedure:

1. A mobile robot with blind navigation system was set up.
2. The mobile robot was allowed to move on the tactile paving after the destination has set up same as the next Experiment 3 which is toilet.
3. The time for mobile robot to reach the destination was recorded.
4. The border markers are located near the turning corner. The number of times of hitting the border makers at the corner is counted. This is to test the functionality of compass coding in programming where need to avoid the corner and made the perfect corner turning.
5. Measurement was repeated about 10 times and observation is noted.
6. The raw data obtained from the repeatability test can be analyzed by calculating the average (mean) and standard deviation.

3.9.4 Experiment 2 Parameters

The performance of the mobile robot navigation is evaluated in some testing parameter. The first parameter is number of times of the mobile robot hits the border marker when take corners. The purpose is to check the deviation or number of times where the robot travels out of the path. The second parameter is the time taken for the mobile robot needed to complete the path. The purpose of using path completion time is to determine the reliability of the shortest path implemented by the A* algorithm. The third parameter is the degrees of angle of the compass when take the corner turning. This is to ensure the corner turning is effective and the angle calculation by the digital compass.

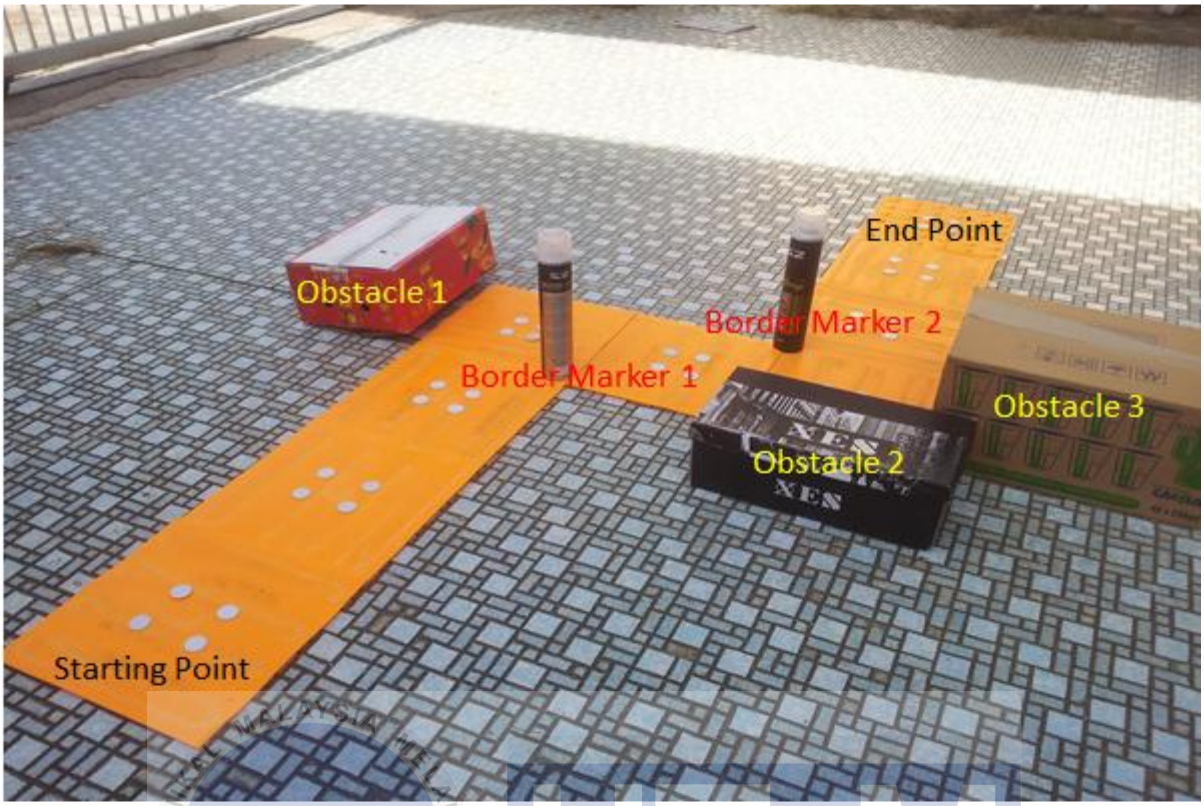


Figure 3.31: Experiment path for mobile robot

Research Question (Experiment 3)

1. Is the blind navigation device guides the visually impaired person with correct walking and turning direction?
2. Is the visually impaired person able to reach the desired destination safety?

Hypothesis:

1. The blind navigation device guides the visually impaired person walks on the straight path and turns the corner.
2. The visually impaired person can reach the desired destination guided by the blind navigation device without hitting the obstacles.

3.9.5 Experiment 3: Lab test of blind navigation device

Objective: To evaluate the performance of navigation device developed in terms of field test and the validity of algorithm.

Materials: Second design blind navigation device, tactile paving, RFID tags, obstacles

In order to evaluate the performance of blind navigation device developed in terms of field test and the validity of algorithm, participants were required to test the blind navigation device on tactile paving. Participants were required to hold the blind navigation device. The users need to key in the destination on the modified keypad with the braille code and wait for the guidance from the headset. At the same time, the blind navigation device was connected to the laptop through wireless communication to obtain the map information. The time used to complete the path was recorded. The data was then collected to indicate the performance of the blind navigation device. The raw data can be analyzed using the statistical method.

The procedure was summarized as below;

Procedure:

1. Participants that took part in this experiment were required to close their eyes or wearing the spectacles with coated black as shown in Figure 3.32.
2. Participants were given 1 minute to adapt the dark condition.
3. Participants were required to hold the blind navigation device and wear the Bluetooth headset along the travel until finish the path as shown in Figure 3.33.
4. The wireless communication (ZIGBEE) between the laptop and blind navigation device was connected.
5. The map system created using Processing 2.0 was connected in order to generate the shortest path and provide the navigation which is shown in Figure 3.35.
6. Participants were required to key in destination using the modified keypad with Braille code as shown in Figure 3.36. The destination chosen for this experiment is toilet.
7. The time used to complete the path from starting point to end point was recorded.
8. Experiment was repeated for about 10 times to obtain the repeatability results.
9. The raw data obtained from the repeatability test can be analyzed by calculating the average (mean) and standard deviation.

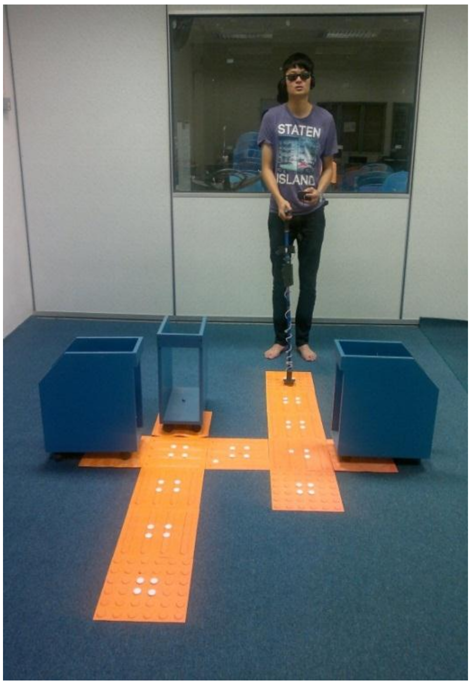


Figure 3.32: Participant with dark spectacle

Figure 3.33: Participant with eye-closed

Figure 3.34:Tactile paving with obstacles

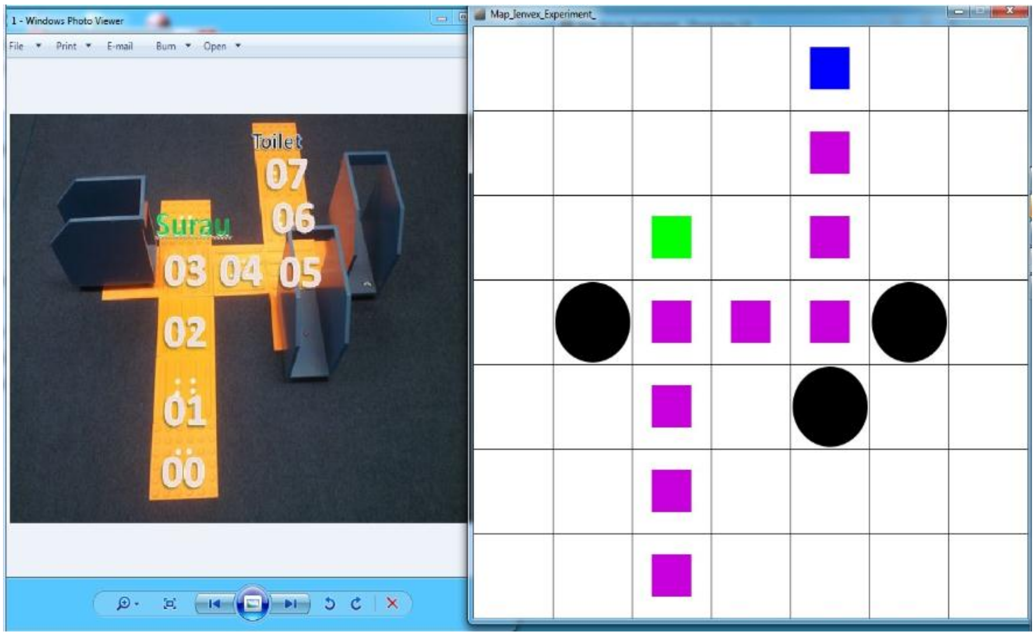


Figure 3.35: Map System created using Processing 2.0

Figure 3.36: Modified Keypad with Braille code

Figure 3.37: Bluetooth headset

3.9.6 Experiment 3 Parameters

The performance of the blind navigation device is evaluated in terms of how much time is needed for the visually impaired person to reach the chosen destination. The path completion time needed for the participant to travel from the start location to the target location is recorded. The purpose of using path completion time is more reliable because the blind navigation system can determine the shortest path with correct direction throughout the whole travelling.

3.10 Summary of Methodology

This chapter explains about the methodology on how to design the hardware, components selection, and experiments designed on how to test the navigation device developed. There are two types of blind navigation device are being developed in this project, however only the second design is the best selection as it fulfills the objectives.

Initially, literature review of the research are studied to gather information related to this project include the RFID technology and navigation system for visual impaired people. Next flow chart of methodology is planned in order to complete the whole project in the predefined time frame.

The following step is to design and develop the hardware. Electronic components such as sensors are studied based on data sheet to determine which ones is suitable to achieve the objectives. Schematic diagram is drawn and the wire connection is determined. Then, the circuit is built based on the schematic diagram.

Next, flow chart for blind navigation system to function is needed to be planned. The developed circuit is needed to be tested based on the workflow. Testing and program compiling is needed to ensure the blind navigation device is ready to enter the experiment phase.

At last but not least, the experiments are designed to test and evaluate the performance of blind navigation device.

CHAPTER 4

RESULTS, ANALYSIS AND DISCUSSION

4.1 Results for Experiment 1

Table 4.1 Compass Accuracy Test Results

Times	Pointing to North (Yes/No)	Degrees (°) North (0/360 °)	Relative Error	Percent Relative Error (%)
1	Yes	356.30 °	3.70	1.0278
2	Yes	356.40 °	3.60	1.0000
3	Yes	356.30 °	3.70	1.0278
4	Yes	355.70 °	4.30	1.1944
5	Yes	358.20 °	1.80	0.5000
6	Yes	357.80 °	2.20	0.6111
7	Yes	357.70 °	2.30	0.6389
8	Yes	357.80°	2.20	0.6111
9	Yes	358.20 °	1.80	0.5000
10	Yes	357.90 °	2.10	0.5833
11	Yes	359.20 °	0.80	0.2222
12	Yes	359.30 °	0.70	0.1944
13	Yes	359.20 °	0.80	0.2222
14	Yes	359.30 °	0.70	0.1944
15	Yes	359.30 °	0.70	0.1944
16	Yes	359.20 °	0.80	0.2222
17	Yes	359.20 °	0.80	0.2222
18	Yes	359.60 °	0.40	0.1111
19	Yes	359.40 °	0.60	0.1666

20	Yes	359.40 °	0.60	0.1666
Average (Mean) for 20 times repeatability, $\bar{x}$		**358.27°**	**Average of Percent Relative Error (%)**	**0.4805%**
Standard deviation, σ		**1.22°**		

Table 4.2 shows the digital compass is always pointing to the North direction. This is very important to ensure that the digital compass functions well and able to determine direction during the navigation purpose. This shows that proposed experiment to test the accuracy of the compass is said to achieve validity.

Standard deviation determines how much variation or deviation from the average. The standard deviation for this repeatability test is 1.22° and this low standard deviation indicates that the degrees are very close to the expected value (mean). The standard deviation is reliable and acceptable because the accuracy of the digital compass measured up to two decimal places which is very high resolution during the changes of degrees.

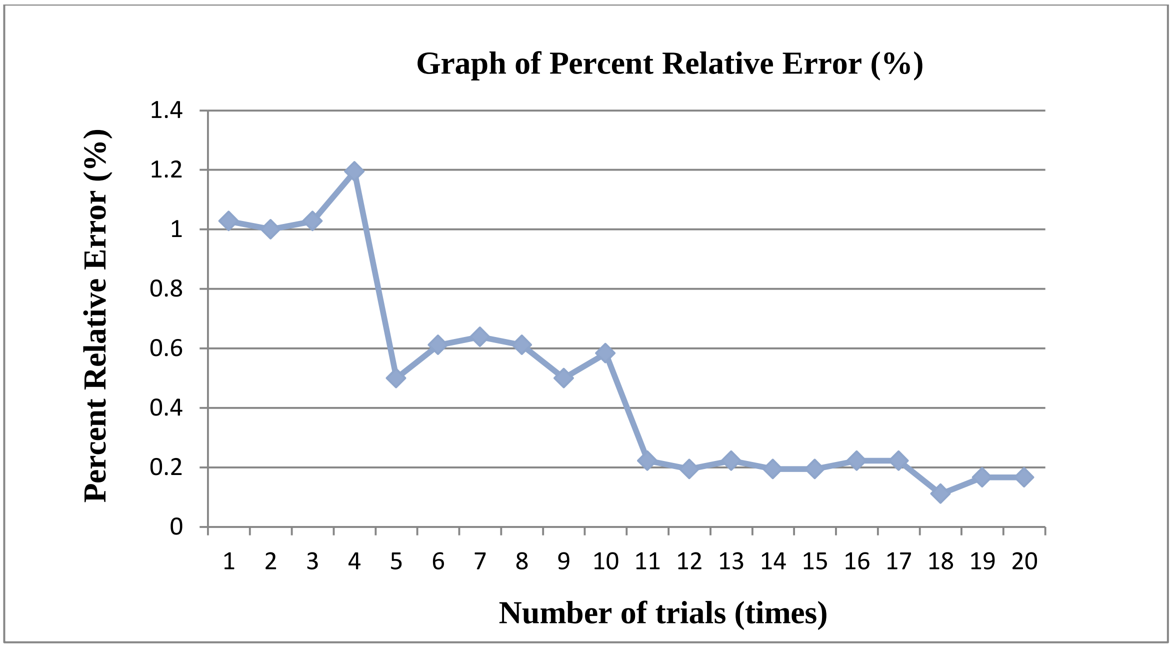


Figure 4.1: Graph of Percent Relative Error (%) vs. number of trials (times)

Figure 4.1 above shows the percent relative error (%) of the readings obtained from the digital compass module when it is pointing to the North. The data obtained from the simulation is the interval data because the data obtained in the continuous way which is the experiment keep repeats in 20 times. The maximum peak value of the percent relative error is 1.1944%. The average percent relative error is 0.4805%.The graph shows the percent relative error is decreasing gradually towards the end and becomes nearly constant between the 11 and 17 times of trials. This implies that the relative error is getting smaller and the readings are very close to the 360° when the digital compass points to the North. The data obtained is said to have high reliability because the percent of relative error is getting smaller after 20 times of repeating test. Besides, the digital compass has very high sensitivity and able to produce significant value at tenth decimal places.

4.2 Discussion for Experiment 1

There are many factors need to be considered during the compass accuracy test experiment. Systematic error is one the most common error where there is bias in measurement lead to average measurements differs from the actual value. Systematic error may occur due to high sensitivity of the instrument (digital compass module) where the reading obtained is non-linear but keep changing when it is pointing to the North.

Besides, the random errors need to be considered as well. Random errors are caused by unpredictable changes in the experiment due to environmental conditions. Environmental effects like magnetic field influence can affect the results of the experiment. The magnetic field can distort the signals to the magnetometer and cause the changes of magnetic pole reading. For example, compass can point to the North because the North end of compass aligns with the Earth's magnetic field and attracts the South poles of earth magnetic field. If there is a North poles magnet placed near to the compass and it will distort the magnetic field and change the reading because same magnetic field charges repel while different charges attract each other.

Human error is another factor that can affect the experiment. Human error means something that has been done but not intended. Pointing the compass by hand is sometimes not very accurate to get the North poles reading. Unless the digital compass is fixed or taped at fixed position and reference is needed such as real compass to ensure the pointing direction.

4.2.1 Limitation of First Blind Navigation Device

The digital compass module HMC6352 can be easily affected by magnetic item. When a magnet is placed near to the magnetometer (digital compass module), distortion happened on the heading output and there is a variation in compass reading. When the magnet is moved away from the digital compass, the reading of the compass does not show the accurate and actual reading but there is a ± 10 degrees deviation. Items like 9V battery and USB cooler pad that have the magnet need to be put away to ensure there is no magnetic field influence. The power supply consumption is another main concern for the navigation device. Power bank is used during the experiment, however there is still need a high power capacity battery which can stands for long hours like portable lithium-ion battery.

Figure 4.2: Compass Reading When Point to North

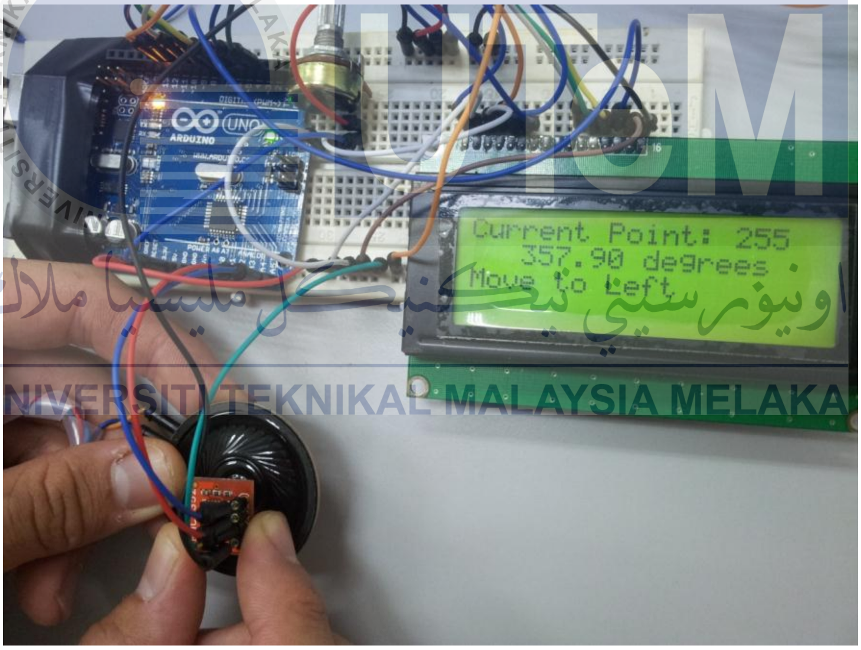


Figure 4.3: Deviation of 7 degrees of the compass reading when a magnet is placed near to it

4.2.2 Precautions for First Blind Navigation Device

There is solution to solve the systematic error. The common method to solve systematic error is calibration of the measurement. Since the digital compass module used in this project is HMC6352 which is compatible with ARDUINO, therefore programming in ARDUINO can used to calibrate the digital compass. To calibrate the digital compass, it is need to enter the User Calibration Mode which is Table below. Command ‘C (43)’ is need to set or declared at the HMC6352SlaveAddress.

int HMC6352SlaveAddress = 0x42 is changed into int HMC6352SlaveAddress = 0x43 and set delay (10000) which is 10 seconds. The digital compass is automatically calibrated by itself after 10 seconds and the digital compass are able to function properly after the calibration.

Table 4.2: Compass Module Data Sheet

Command Byte ASCII (hex)	Argument 1 Byte (Binary)	Argument 2 Byte (Binary)	Response 1 Byte (Binary)	Response 2 Byte (Binary)	Description
w (77)	EEPROM Address	Data			Write to EEPROM
r (72)	EEPROM Address		Data		Read from EEPROM
G (47)	RAM Address	Data			Write to RAM Register
g (67)	RAM Address		Data		Read from RAM Register
S (53)					Enter Sleep Mode (Sleep)
W (57)					Exit Sleep Mode (Wakeup)
O (4F)					Update Bridge Offsets (S/R Now)
C (43)					Enter User Calibration Mode
E (45)					Exit User Calibration Mode
L (4C)					Save Op Mode to EEPROM
A (41)			MSB Data	LSB Data	Get Data. Compensate and Calculate New Heading

```
sketch_nov05a   compass §
#include <Wire.h>// initialize the library with the numbers of the interface pins
int HMC6352SlaveAddress = 0x42;
int HMC6352ReadAddress = 0x41; //"A" in hex, A command is:
int headingValue;
void setup(){
  HMC6352SlaveAddress = HMC6352SlaveAddress >> 1;
  Serial.begin(9600);
  Wire.begin();
}
void loop(){
  //"Get Data. Compensate and Calculate New Heading"
  Wire.beginTransmission(HMC6352SlaveAddress);

 Wire.write(HMC6352ReadAddress);
  Wire.endTransmission();
  delay(5000);

  Wire.requestFrom(HMC6352SlaveAddress, 2); //get the two data bytes, MSB and LSB
  byte MSB = Wire.read();
  byte LSB = Wire.read();

  float headingSum = (MSB << 8) + LSB; //(MSB / LSB sum)
  float headingInt = headingSum / 10;

  Serial.print(headingInt);
  Serial.println(" degrees");
  delay(100);
}
```

Figure 4.4: Coding For Compass Calibration

4.2.3 Performance of the first blind navigation device

The first navigation device developed has been tested for its performance. The navigation device is built with compass for direction guidance and a voice module with speaker able to inform user about direction.

The high accuracy property of digital compass is able to give direction fast and accurately. Figure 4.5 below demonstrate when user go forward towards North direction at the starting point, the RFID reader read the tags and digital compass measure the direction and display the direction on the LCD display and inform user to go straight through the speaker.

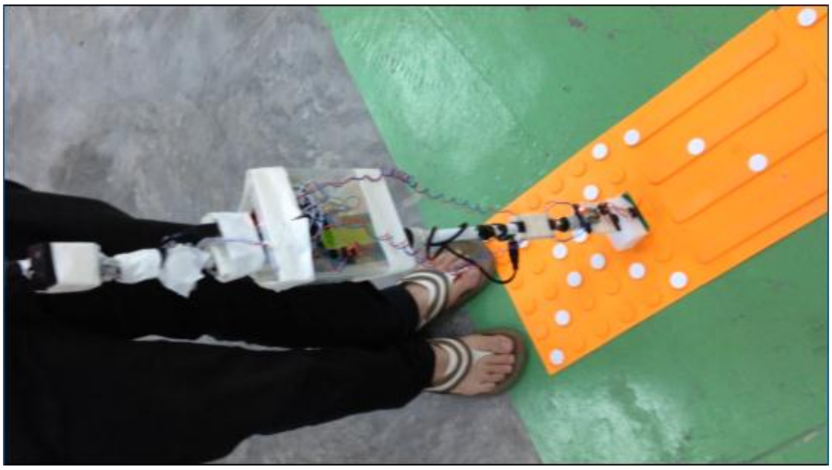


Figure 4.5: Start Point

Figure 4.6: Text Display on LCD

When the user reach a 90 degree junction, RFID reader reads the tags first and then digital compass will determine the direction where the route should be taken. After that, voice module will release "90 degree Turn Left" audio signal and guide the user to take the left hand side of the junction. After passing through the junction, user continues travel with the aid of the navigation device.

The digital compass will able to detect the error if the user travels out of the direction. and misdirection may lead to wrong path. Thus the voice module will release "warning" audio signal repeatedly and alert user from taking the wrong path.

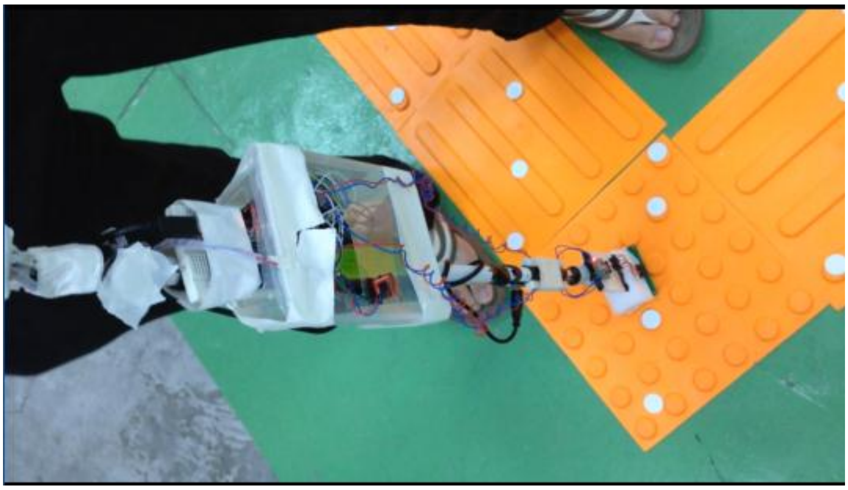


Figure 4.7: 90 degree Junction

4.3 Results for Experiment 2

In this experiment, there are three variables need to be considered but the variables are carried out separately. There are number of times hitting the border marker, degree of angle of compass when mobile robot take the corner and time needed for mobile robot to complete the path. For this case, the constant variables of the experiment are degree of angle of compass and time needed to complete then path. The responding variable for this case is number of times hitting the border marker.

Table 4.3: Number of times hitting the border marker

Times	Number of times hitting the two border marker
1	2 hits(1st border, 2nd border)
2	2 hits(1st border, 2nd border)
3	2 hits(1st border, 2nd border)
4	1hit (1st border)
5	1hit (1st border)
6	1hit (2nd border)
7	1hit (1st border)
8	1hit (2nd border)
9	1hit(2nd border)
10	1hit(2nd border)

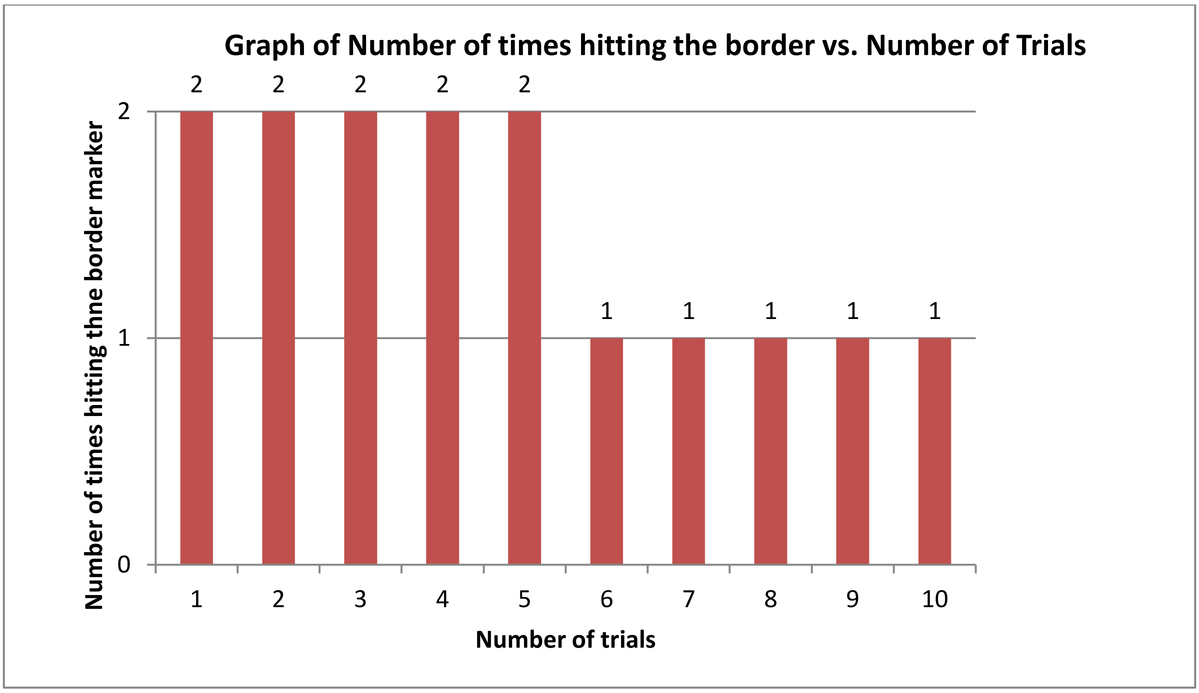


Figure 4.8: Graph of number of times hitting the border vs. Number of trials (times)

Figure 4.8 shows the graph of number of times hitting the border vs. number of trials. The highest number of times of hitting the border is 2 times where the rests are only 1 times. There are different kinds of number of hitting; some are two times hitting the first and second border, some are once times hitting the first border and some are once times hitting the second border. There are two hitting of border during the first 5 trials, followed by 1 hitting for the next 5 times. The hitting to the border implies that the range setting for angle comparison in the digital compass's coding part must be corrected. The border marker is important to ensure the mobile robot take the corner effectively by not hitting on it. This data parameter is crucial because the programming will be then implemented later on the blind navigation device and to ensure the safety of the visual impaired person when they are turning the corner by not hitting the obstacles. The programming need to be adjusted and compiled many times to ensure the safety corner turning. As it can see from the graph of Figure 4.8 at the end there is no hitting the border it implies that the coding is corrected and ready to implement at the blind navigation device.

For this case, the constant variables of the experiment are degree of angle of compass and number of times hitting the border marker. The responding variable for this case is time taken by mobile robot to complete the path.

Table 4.4: Time taken to complete path (seconds) by mobile robot

Times	**Time taken to complete the path (seconds)**
1	33
2	32
3	33
4	33
5	32
6	33
7	32
8	31
9	32
10	33
Average(Mean) for 10 times repeatability, $\bar{x}$	32
Standard deviation, σ	10.11

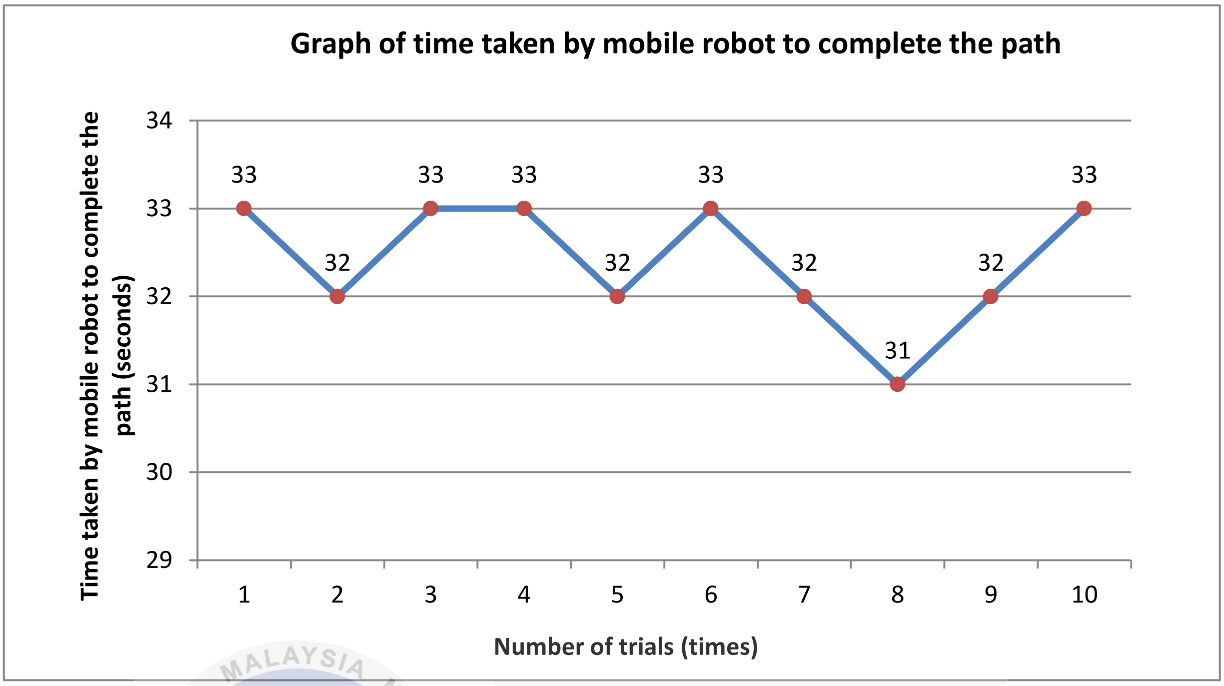


Figure 4.9: Graph of time taken to complete the path (seconds)

Figure 4.9 shows the graph of time taken to complete the path (seconds) vs. number of trials. The average time for the mobile robot to reach the destination is 32 seconds. The maximum peak value of the path completion time is 33 seconds while the minimum peak value is 31 seconds. The standard deviation is 10.11 seconds. The value is reliable because the experiment repeats about ten times. The graph shows that there are indeed some variations between 10 times of repeatability. This is because there are differences in time when the RFID reader reads the tags. The total path completion time is equal the time for the RFID reader to read the tags and the time needed to take the complete corner turning. This is because every time the RFID reader takes time to read the encoded information and interpret the information. Regardless of how, in the end the coding should be compiled correctly for the mobile robot to reach the destination around average 32 seconds to ensure algorithm is precise and time-saving for travelling as it will implemented later on the blind navigation device for the evaluation.

For the third case where the experiments is still the same; the constant variables of the experiment are time needed for mobile robot to complete the path and number of times hitting the border marker. The responding variable for this case is degree of angle of the compass when mobile robot turn the corner.

Table 4.5: Degrees of angle of the compass when mobile robot take the corner turning

Times	**Degrees of angle of the compass when mobile robot turn the corner [rounded off value(°)]**
1	88
2	87
3	88
4	88
5	87
6	89
7	88
8	87
9	89
10	88
Average(Mean) for 10 times repeatability, $\bar{x}$	88
Standard deviation, σ	27.79

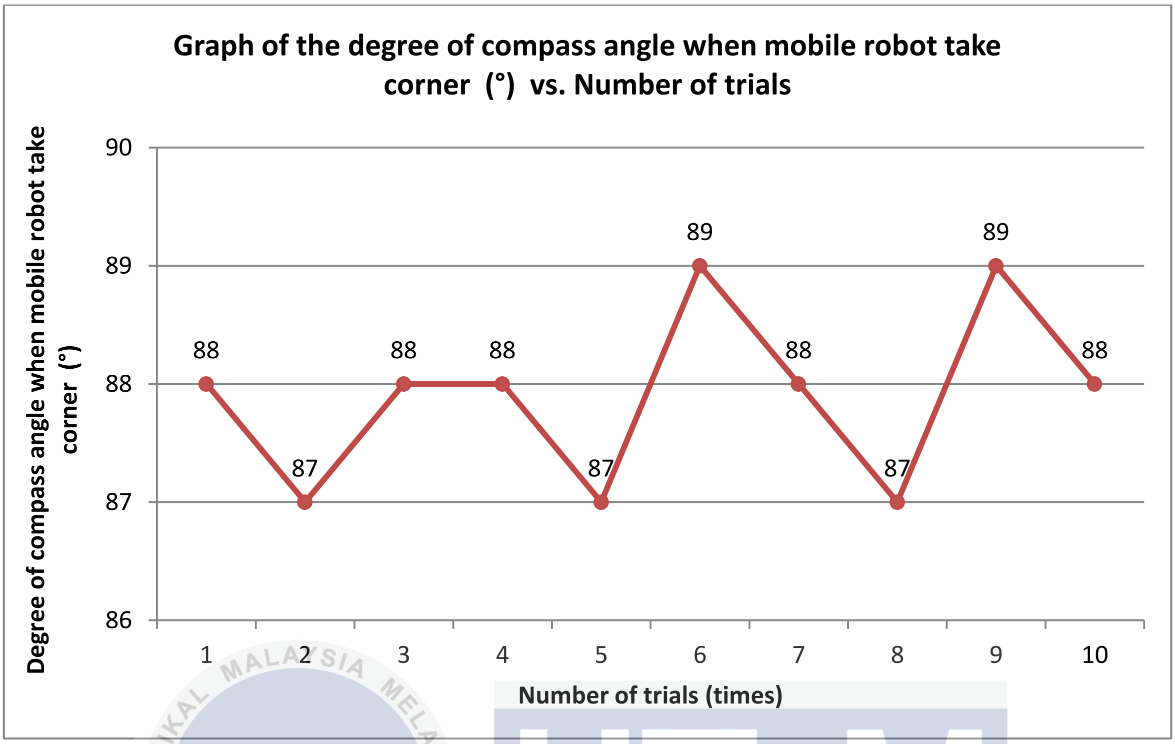


Figure 4.10: Graph of degrees of compass when mobile robot take corner (°) vs. number of trials (times)

Figure 4.10 shows the graph of degrees of angle of compass when take corner (°) vs. number of trials. The graph is like a step graph where the degree of angles is within the range from 87(°) to 89 (°). The largest deviation from 90(°) is only 3 degrees or 3.33% which is 87(°).The average deviation is 88(°) .The standard deviation is 27.79(°). The value of standard deviation is reliable because the deviation of the mobile robot from the straight path is not so large and it also implies that the mobile robot is able to take corner accurately but not precise during many experiments trials. Furthermore, there is not overshoot for the angle deviation, e.g. 90(°) and above. The means the phenomenon of mobile robot is sliding out of the path is not usually happened. This experiment's parameter is important to ensure that the compass coding in the programming is reliable and less debugging error because it will later implemented on the second design of blind navigation device so that the digital compass is able to provide accurate turning angle when turn left and right.

4.4 Discussion for Experiment 2

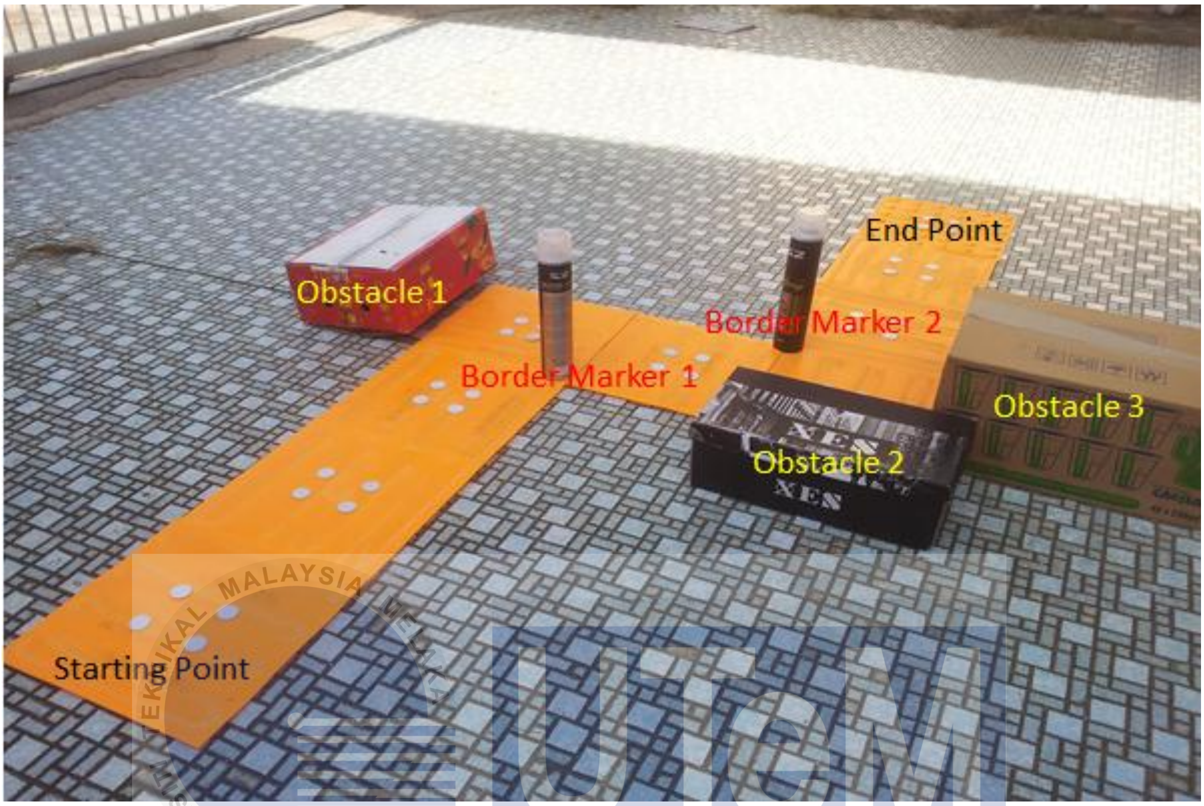


Figure 4.11: Path for mobile robot

Figure 4.11 above shows the setup for the experiment 2. There is a travel path for mobile robot which is 210cm total in length. The path is same as the Experiment 3 where the participants will travel on it with the second blind navigation device. There are 3 obstacles which cannot pass through and 2 cylinder-shaped border markers. The function of both the border markers is to check deviation of the mobile robot and determine how many times the robot travels out of path during the locomotion of the whole tactile paving. The first border marker is to check the corner turning when the mobile robot is turning right and the second border marker is to check the corner turning when the mobile robot is turning left.

4.4.1 Influence factors for Experiment 2

There are some factors might affect the results of the experiment and need to be aware of. Among them are the random errors and human error.

Random error is caused by unpredictable changes due to environmental conditions. For this case, the experiment set up for the tactile paving is on the tile surface. The path completion time might be affected because the uneven road surface may delay the time for the mobile robot to reach the destination.

Furthermore, the path completion time also might be affected by human errors. The human error is the timing to switch on the mobile robot whereby every time to switch on is different.

4.4.2 Performance of the mobile robot with the A* algorithm

Figure 4.12: Mobile robot takes right corner

Figure 4.12 shows the mobile robot move on the tactile paving. A mobile navigation system for the visually impaired person was being tested for the validity of the A star algorithm. The mobile robot consists of ARDUINO Pro Mini microcontroller, digital compass and RFID reader. Once the destination had been set, the mobile robot is required to travel on the tactile paving itself. The system is then implemented on the electronic cane. Figure above shows the mobile robot passes the first corner to the right hand side. The mobile robot takes time for about 3 seconds to complete the right corner turning.

Figure 4.13: Mobile robot takes left corner

Figure 4.13 shows the mobile robot passes the second corner to the left hand side. The mobile robot takes about 3 seconds to complete the left corner turning.

In order to avoid from hitting the border marker, the digital compass will always compare the angle and guide the navigation. The angle of corner turning must be totally equal to the 90 degree of the compass as shown in Figure x below. Tolerance (±2 degree) is still can be accepted. This step is crucial and critical because this navigation algorithm will be then implemented in the blind navigation device to ensure the personal safety of the visually impaired person during corner turning by avoiding the obstacles.

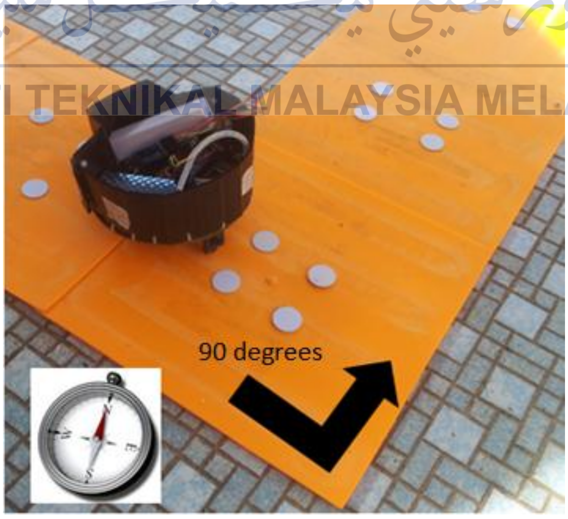


Figure 4.14: Angle determined by digital compass when corner turning

Navigation at the indoor unknown environment is the biggest challenges for the visual impaired person. The navigation system available is not capable to provide the precision due to localization method.

The testing of shortest path algorithm on the mobile robot is much easier than human because human have much complex movements than mobile robots. They differ with respect to the type of body movements and the degree of freedoms. Data acquisition is much harder compared than mobile robot.

The benefit of testing the validity of A* algorithm on the robot is saves time and able to get the accurate results immediately on the spot. Besides, many trials can be done to compile the programming error.

4.5 Result for Experiment 3

Table 4.6: Time taken to complete the path (seconds)

Times	**Time taken to complete the path (seconds)**
1	48
2	47
3	48
4	48
5	47
6	48
7	48
8	46
9	47
10	48
Average(Mean) for 10 times repeatability, $\bar{x}$	47
Standard deviation, σ	14.9

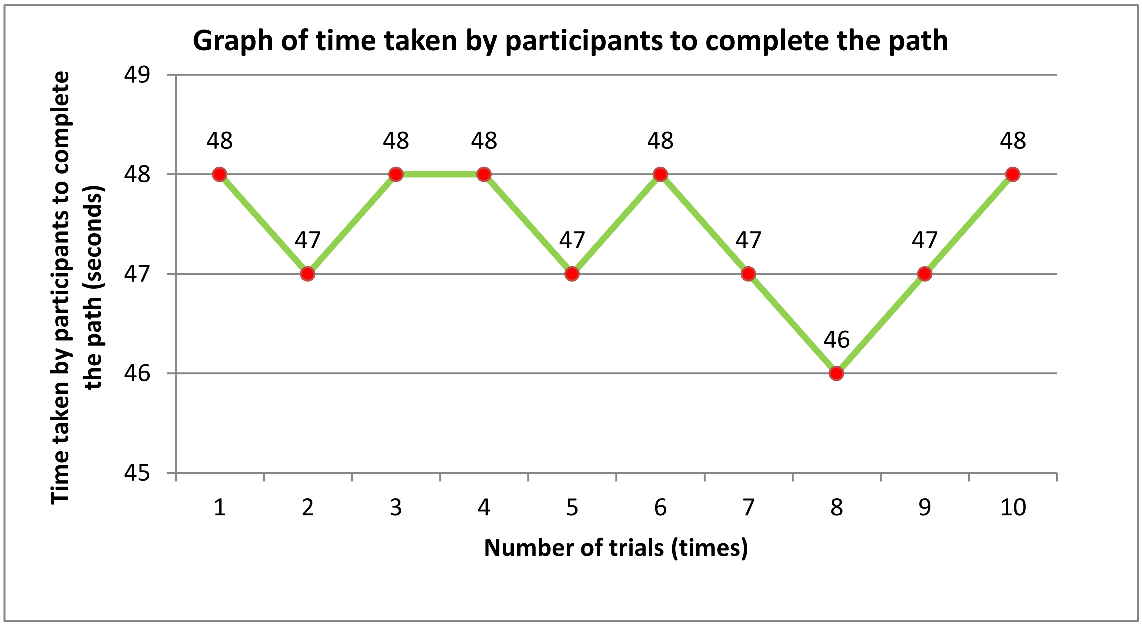


Figure 4.15: Graph of Time taken to complete the paths (seconds) vs. number of trials (times)

Figure 4.15 above shows the time taken to complete the paths. The raw data is interval type and the average time is about 47 seconds after 10 times of repeatability. The maximum peak value of the time is 48 seconds while the minimum peak value is 46 seconds. The standard deviation is 14.9 seconds. The standard deviation value is reliable and there is not much deviation in the time frame. The graph shows that there are differences in time taken to complete the path. This is because the voice commands received every time to the participants for the experiment is different and there is some delay in sending voice commands. The walking action is totally dependent on the time of receiving the commands. It means the participant walks first if he or she had received the voice commands .Thus, the time taken to complete the path is actually the combination of the receive commands time and action time. The result above is reliable because the time tolerance is only ± 2 seconds and proves that the blind navigation device is able to guide visually impaired person. This pattern of this time graph is almost same as the time graph to complete the path by mobile robot. The only difference between these two graphs is only the time between the mobile robot and participants is 15 seconds to complete the path.

Table 4.7: Comparison between time to complete the path using RFID network and time to complete the path using RFID network plus A* algorithm

Times	Time to complete the path using RFID network (seconds)	Time to complete the path using RFID network plus A* algorithm (seconds)
1	58	48
2	57	47
3	58	48
4	58	48
5	57	47
6	57	48
7	56	47
8	57	46
9	57	47
10	58	48
Average(Mean) for 10 times repeatability, $\bar{x}$	57	47
Standard deviation, σ	18.0	14.9

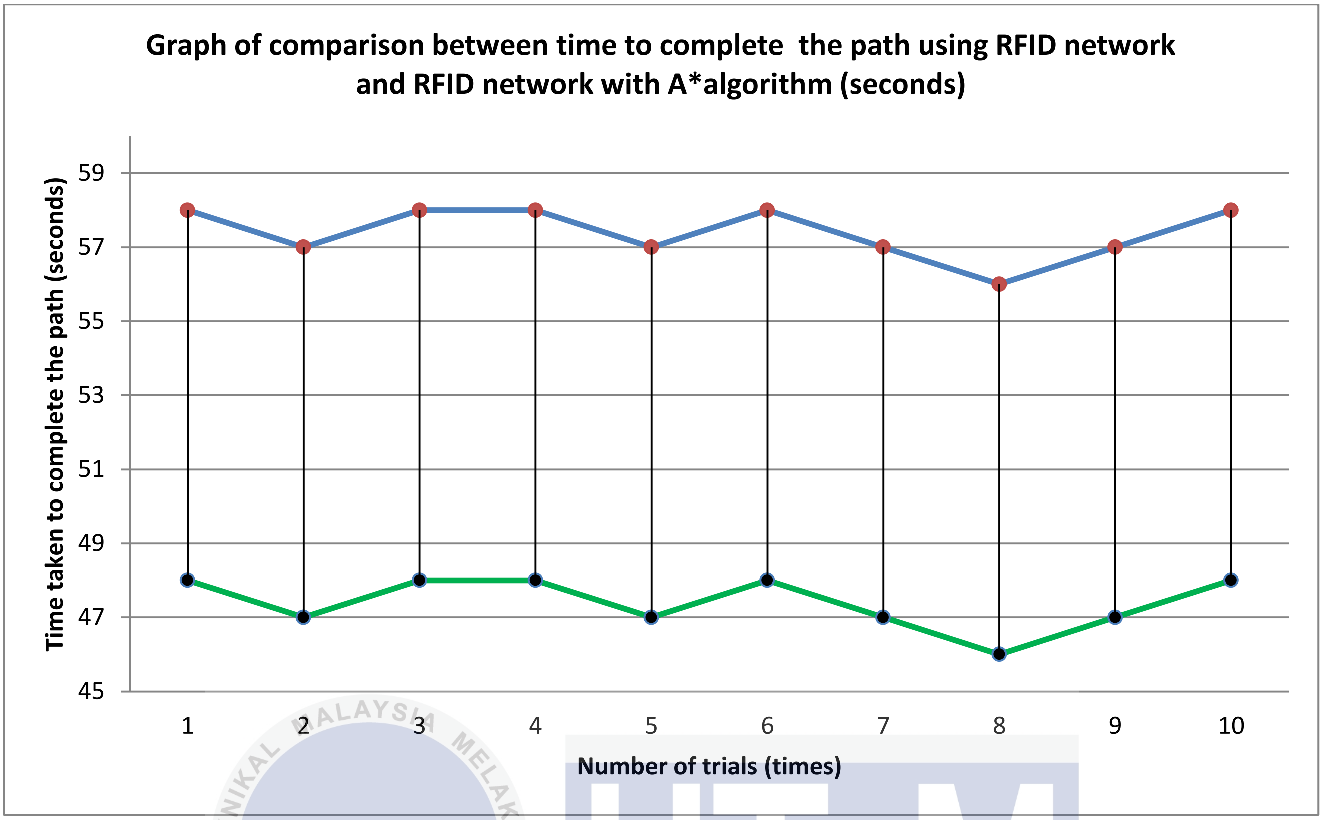


Figure 4.16: Graph of comparison between time to complete the path using RFID network and RFID network with A*algorithm (seconds)

Figure 4.16 shows the graph of comparison between time to complete the path using RFID network and RFID network plus A* algorithm. The red column represents the time to complete the path using RFID network while the blue column represents the time to complete the path using RFID network plus A*algorithm. The average time used to complete the path using RFID network is 57 seconds while the average for RFID network and A* algorithm is 47 seconds. The maximum time using RFID network is 58 seconds and minimum time is 46 seconds. On the other hand, the maximum time using RFID network and A*algorithm is 48 seconds and the minimum time is 46 seconds. As the data shown, the difference from the total time taken by using RFID network with A* algorithm is 10 seconds from the method using RFID network. The A* algorithm is proved to save times on finding the destination point rather than participants explore by themselves since it can determine the shortest path once the starting and destination point is known.

4.6 Discussion for Experiment 3

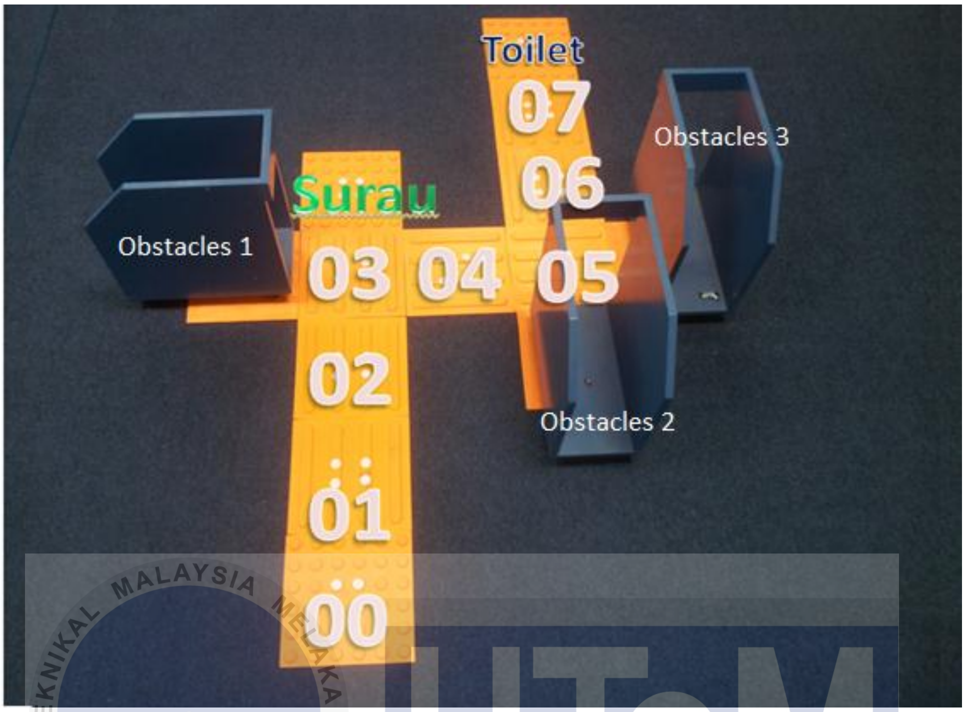


Figure 4.17: Tactile paving with numbered position and target destination

Figure 4.16 shows the tactile paving used for the experiment. The tactile paving that numbered with 00, 01, 02, 03, 04, 05, 06, 07 are the points paths while blue objects represent the obstacles that is unable to pass through. There are two destinations on this experiment which are prayer room and toilet.

4.6.1 Influence factor in Experiment 3

There is some influence factors to be considered during this blind navigation evaluation test. Systematic error and human error is to be discussed. Systematic error where there is bias in measurement lead to the path completion time. The error occurred for the time response of the system where there is some time delay for the ZIGBEE during data transmission and sending voice commands.

Human error is another error that is not intended and cannot avoided just like the Murphy's Law that stated as: "Anything that can go wrong, will go wrong". For this case, the human error is response of the participants when they start to walk when they received the voice commands. There are some delay at the starting point and corner turning. This will give the different results for the path completion time. However, the time that is needed to complete the path does not emphasize too much on how fast the person reach the destination, but it is used to prove that the A* algorithm indeed can determine the shortest path for them to reach the target location which is considered the greatest benefits for the visual impaired person.

4.6.2 I^2C Connection Between Modified Keypad with Braille Code and Voice Module

I^2C is a serial single –ended computer bus that usually used to connect two ARDUINO microcontrollers together at pin A4 and A5. There is I^2C connection in the second blind navigation device whereby the modified keypad is connected to an ARDUINO Pro Mini and then is linked with another ARDUINO Pro Mini which is shown in Figure 4.17. However, the set up for the connection is important. If the keypad is not functioning, then the whole programming for the navigation system is at halt and lead to no voice commands is being sent to the user later. The better way to ensure the good connection is to tap the multi meter at pin A4 and A5 to test whether is connected or not with pen probes.

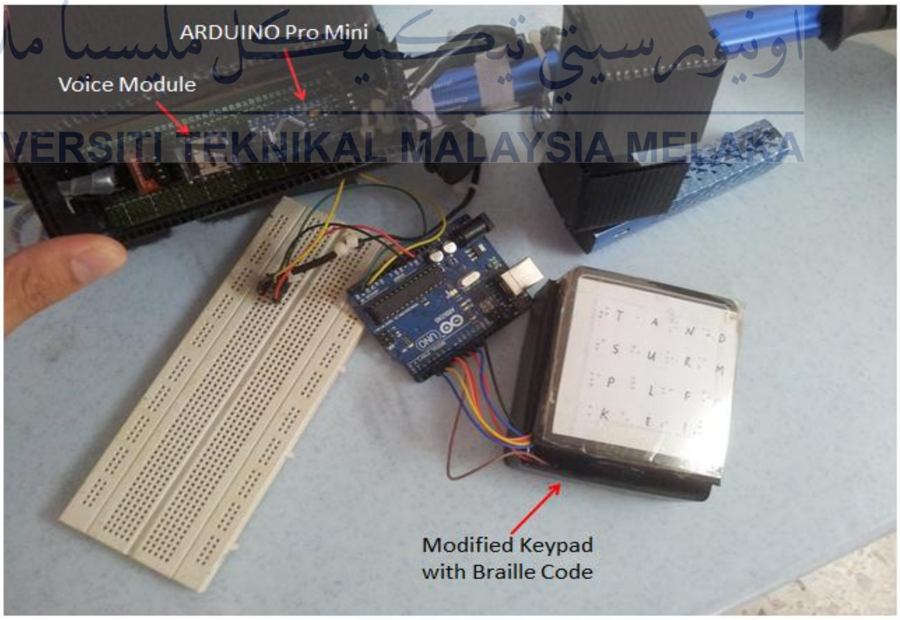


Figure 4.18: I^2C Connection of Modified Keypad with Braille Code and Voice Module

4.6.3 Performance of second design navigation device

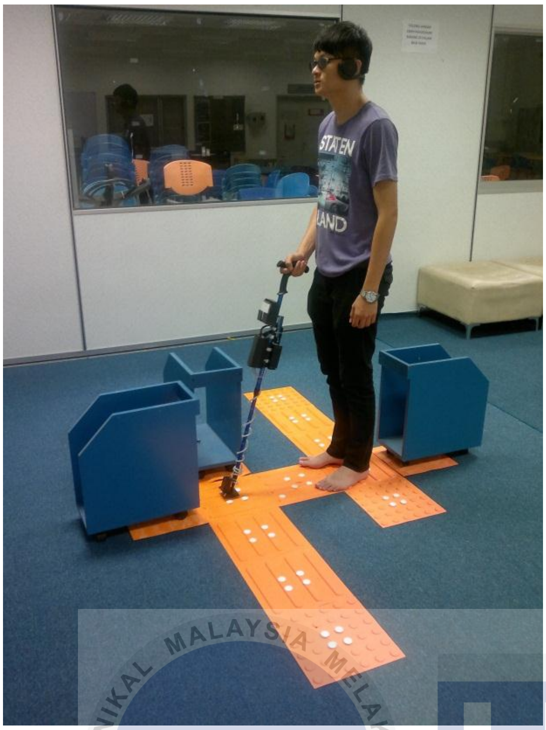


Figure 4.19: Corner turning

Figure 4.20: Reach destination

The second design is the map system from Processing 2.0 integrates with the RFID technology. Once the RFID reader reads the tag, the current location will be sent to the map system. Then user will key in the desired destination. For example, user key in "TANDAS".

The map system that possesses the A* algorithm will determine the shortest path between the current location and end point which is total 7 steps counting from the start point. Since the user still detect the RFID tags, the voice commands will say "Jalan Ke Depan".User then follow the instruction from voice commands and move until the electronic cane contact the new point (tag).

When user reaches 90 degree junction, the digital compass plays important role in calculating the angle and ensure the user is completely turn the corner. For example, the user reaches the L-junction and supposes to turn the corner, the digital compass compares until reach 90 degree. The illustration is shown as Figure 4.21. The voice system will say "Pusing ke Kanan" if need to turn right or "Pusing ke Kiri" if need to turn left.

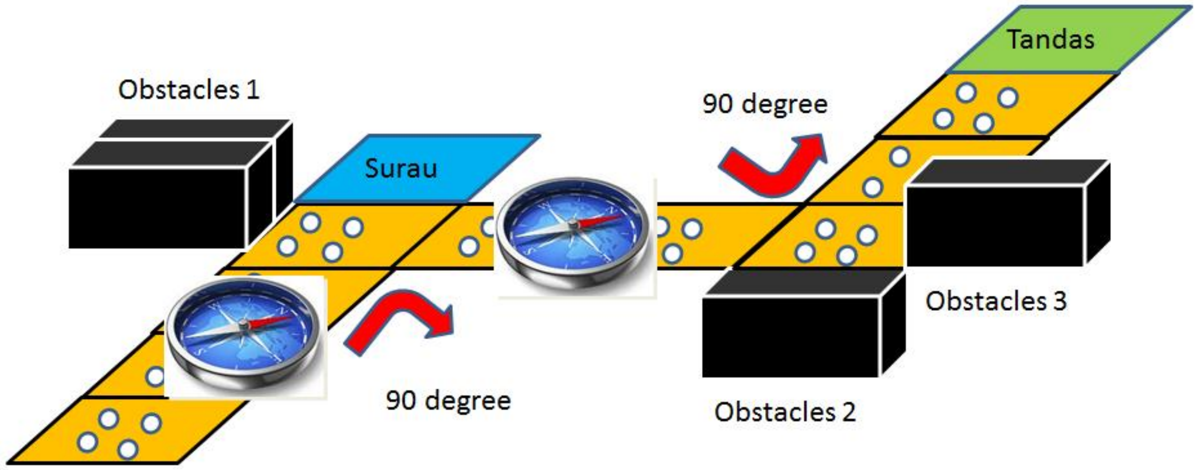


Figure 4.21: Illustration of the corner turning with digital compass

Figure above shows the function of digital compass used in the second design. Digital compass is used to compare the direction where the visual impaired person heads to.

The sample coding is being used in the blind navigation device and mobile robot. It shows the compass is activated to compare the angle when taking the right corner as shown in Figure 4.22. The compass will compare the value between 0 and 85(°) or the value more than 95 (°) and less than 359 (°) which is shown in Figure 4.20. If the condition is matched, then the voice commands will inform to turn right.


```
if(new_state==2)
{
  while(Compass_val<85 && Compass_val>0 || Compass_val>95 && Compass_val<359)
  {
    compass_hmc();
    analogWrite(motor1Pin1 , 0);
    analogWrite(motor1Pin2 , 255);
    analogWrite(motor2Pin1 , 255);
    analogWrite(motor2Pin2 , 255);
    //compass_hmc();
    Serial.println(Compass_val);
    Serial.println("Right but Out of Boundary");
  }
  user_y++;  // Increase user's y coordinate by 1 to indicate turn and walk to right handside
  Listen2:
  cardData();
  Serial.println("Right");  // Tell user to turn and walk to right
```

Figure 4.22: Compass's coding when turns right

```
if(new_state==4)
{
  while(Compass_val<265 && Compass_val>0 || Compass_val>275 && Compass_val<359)
  {
    compass_hmc();
    analogWrite(motor1Pin1 , 0);
    analogWrite(motor1Pin2 , 255);
    analogWrite(motor2Pin1 , 255);
    analogWrite(motor2Pin2 , 255);
    //compass_hmc();
    Serial.println(Compass_val);
    Serial.println("Left but Out of Boundary");
  }
  user_y--;  // Decrease user's y coordinate by 1 to indicate turn and walk to left handsid
  Listen4:
  cardData();
  Serial.println("Left");  // Tell user to turn and walk to left
```

Figure 4.23: Compass's coding when turn left

The sample coding shows the compass is activated to compare the angle when taking the left corner as shown in Figure 4.23. The compass will compare the value between 0 and 265(°) or the value more than 275 (°) and less than 359 (°). If the condition is matched, then the voice commands will inform to turn left.

4.7 Workflow of the Map System with the Second blind navigation device

Figure 4.24 Map Serial Monitor

Figure 4.24 shows the serial monitor of the map system. Once the ZIGBEE wireless network is connected between the laptop and the second blind navigation device, the serial monitor at the bottom of the Processing 2.0 will show the number 255 repeatedly. It means that the RFID tags not yet detected and they are trying to connect to each other wirelessly which is shown in Figure 4.25. It is very natural when write the program code, usually either 1 or 0 will understand by computer. In this case, when the no tags are detected it coded as 255 value, while on the other hand, if the 0x01 tag is detected it will shows the number 1 instead of 255.

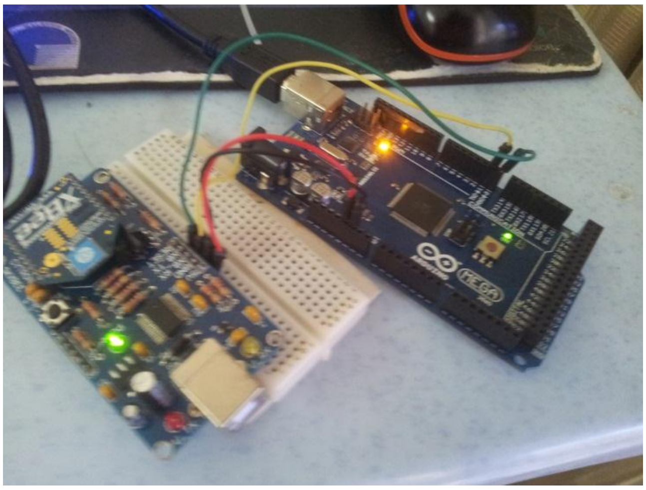


Figure 4.25 ZIGBEE waits for pairing connection

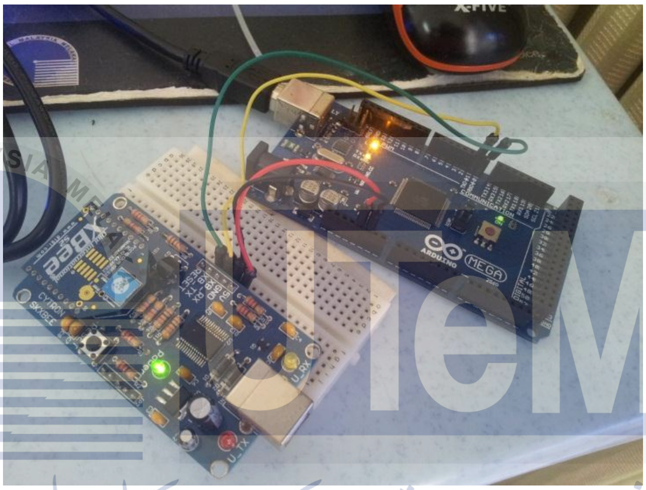


Figure 4.26 ZIGBEE is successfully connected

Figure 4.26 shows the ZIGBEE is successfully connected between the laptop and the blind navigation device. There are two led lights up which is shown at figure 4.24 while there is only one led lights up when there are trying to connect with each other.

When the 0x01 tag is scanned, the serial monitor of Processing 2.0 will shows 1which is shown in Figure 4.27. At the same time, there is a grey circle appears on the windows which is the map GUI (graphical user interface) which is shown in Figure 4.28.

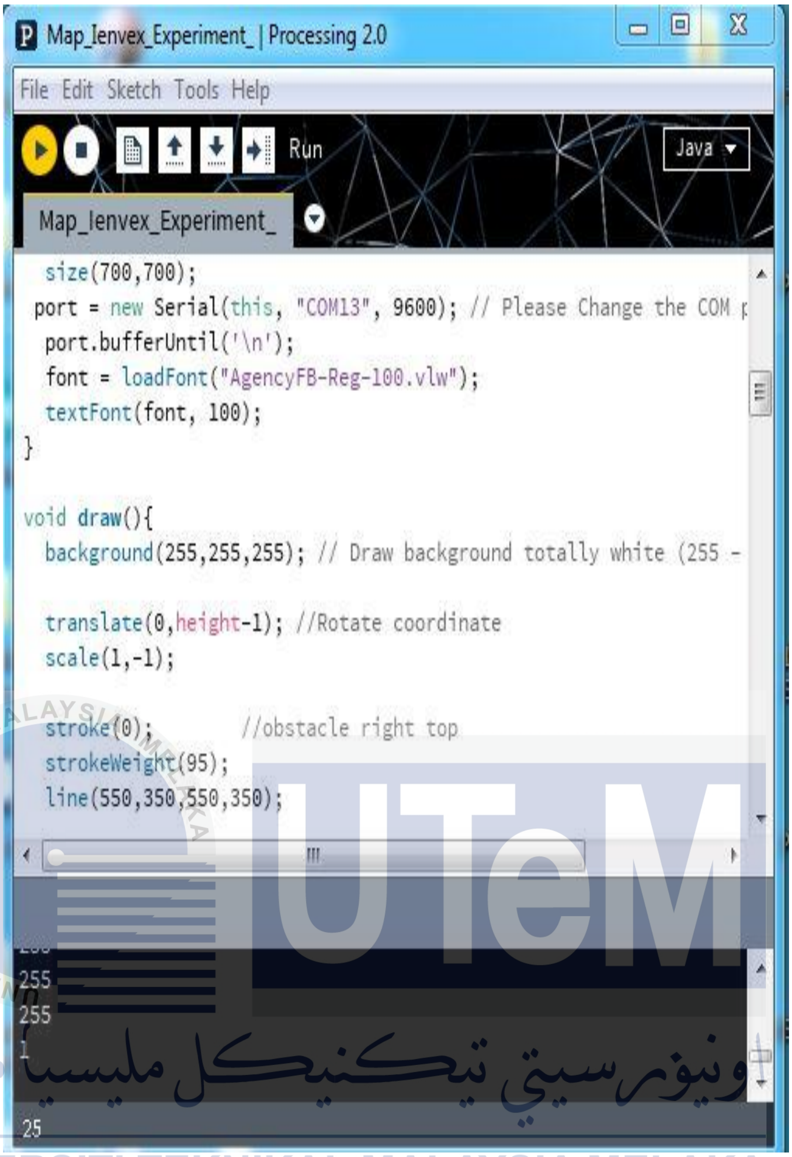


Figure 4.27 Serial monitor shows 1 when 0x01 tag is scanned

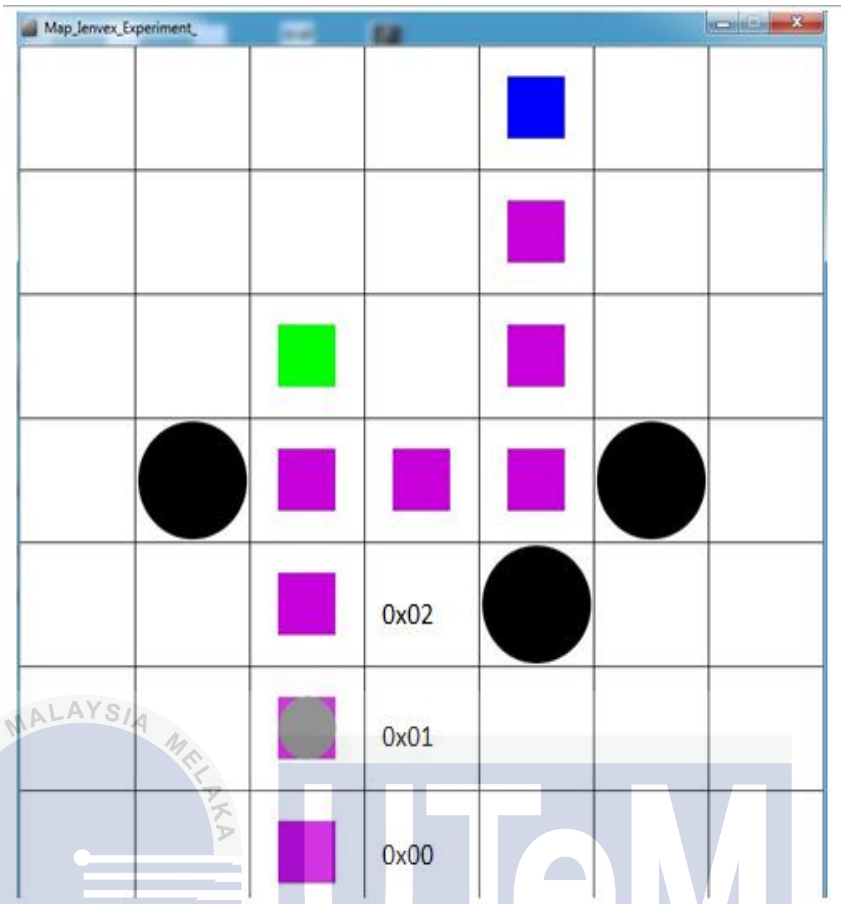


Figure 4.28 The grey circle appear on map when 0x01 tag is scanned

4.8 Map system and A* algorithm

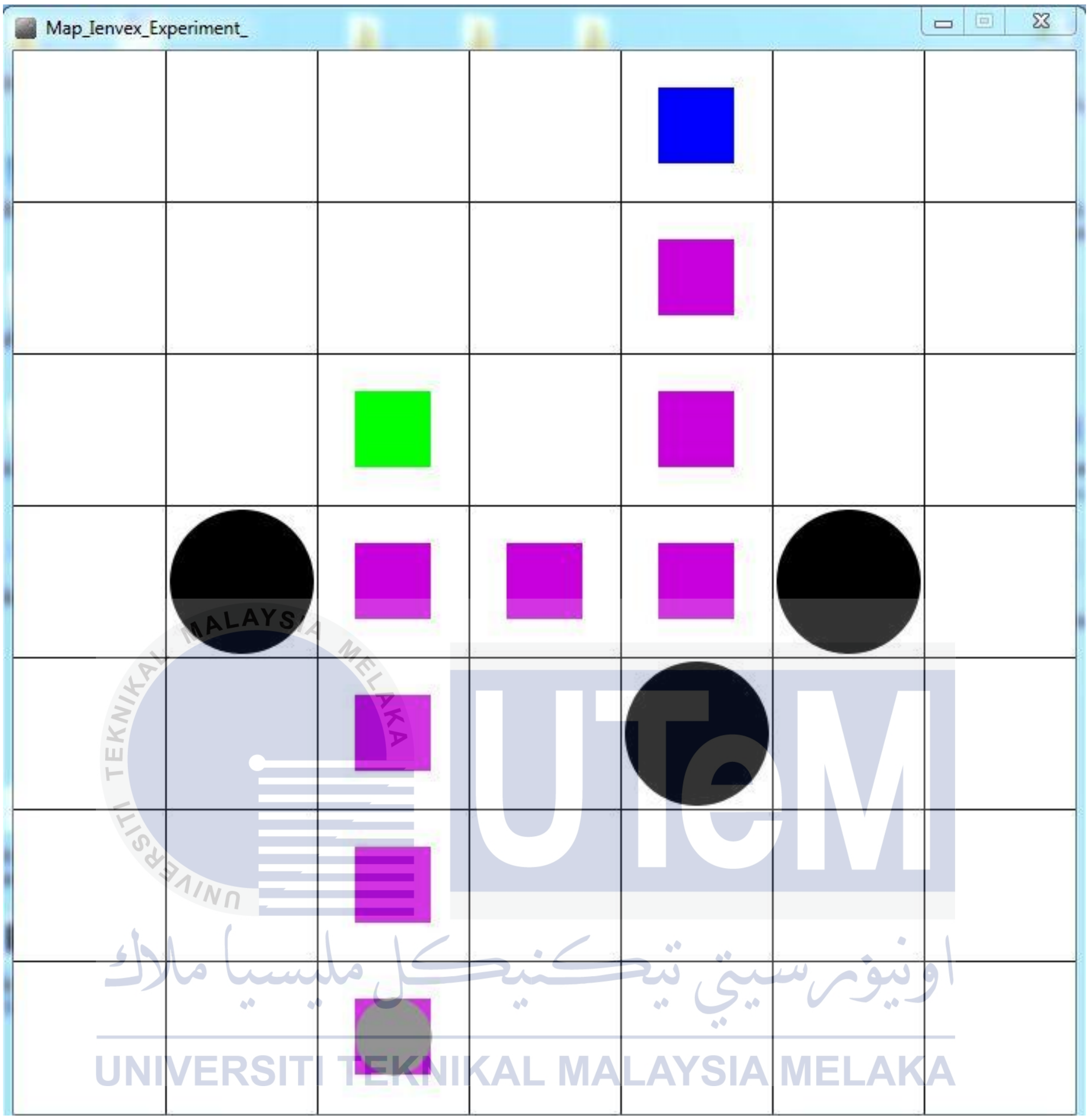


Figure 4.29: Map system by Processing 2.0

Figure 4.29 shows the map system created by Processing 2.0 software. The map system is designed same as the arrangement of the floor plan for Experiment 3. The purple square represents the tactile paving, the green square is the destination "SURAU" and the blue square is the destination "TANDAS". There are three large dark circles which are the obstacles that user cannot pass through.

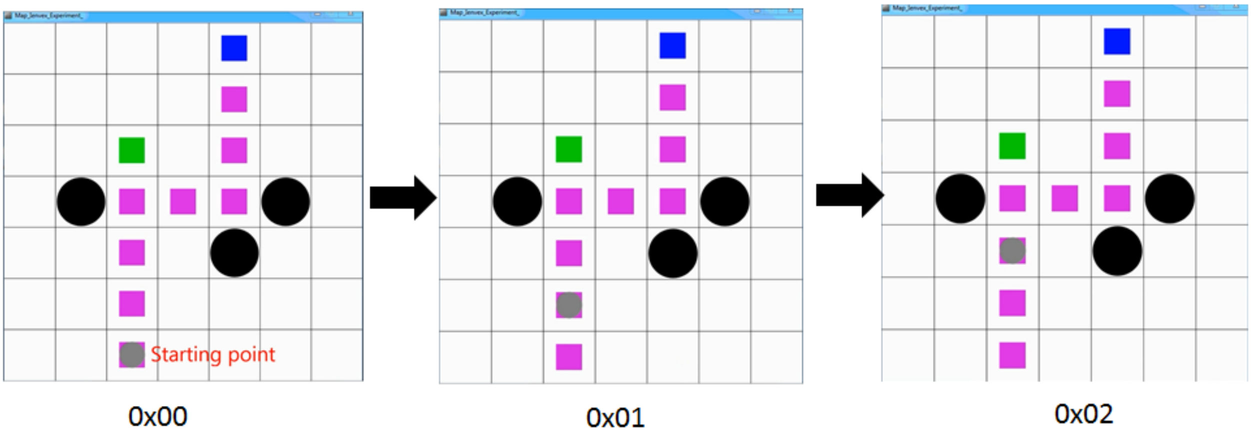


Figure 4.30: Detection of RFID Tags by map system

Figure 4.30 shows the detection of RFID tags by map system. Once the RFID reader reads the tags, a grey circle will be shown on the map system automatically because the RFID detection system had already connected to the laptop through ZIGBEE network wirelessly. For example, tag numbered 0x00 is the starting point, whereas tags 0x01 is the next tag to be scanned and the rest respectively.

Below is the A*algorithm map system created by Processing which is shown in Figure 4.31 and Figure 4.32.

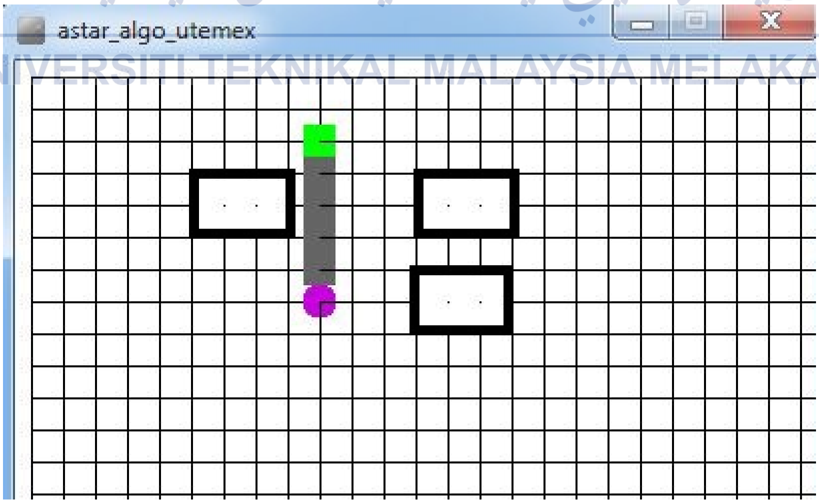


Figure 4.31: Shortest path for the “SURAU”

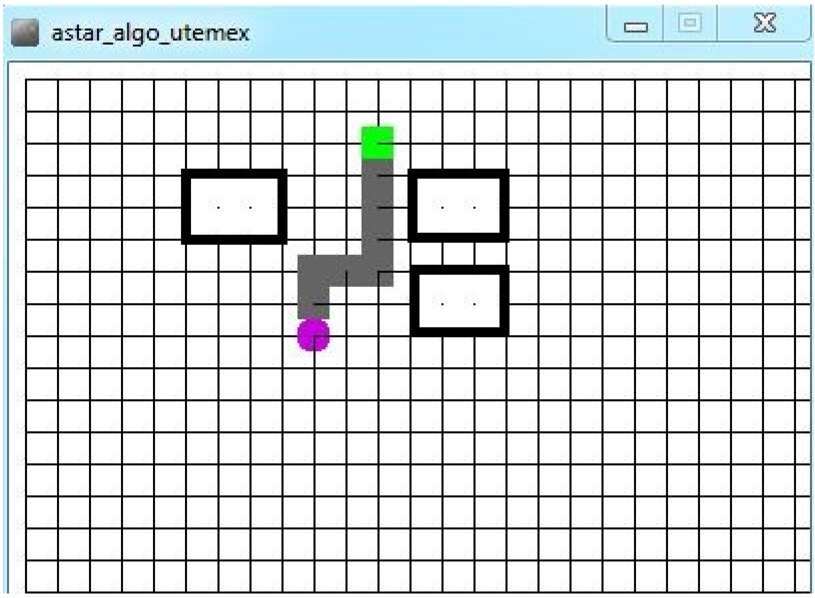


Figure 4.32: Shortest path for “TANDAS”

Figure 4.31 and 4.32 shows the map system processing for the A* algorithm. It can be synchronized together with the map system which is Figure 4.29 to provide the shortest path and how the visually impaired people travel the path. This algorithm is used to find path from start node to the end node using graph and chose the suitable shortest path. For the Figure 4.32, when the tag reader scans the RFID tags where the user at the origin, the initial starting point (purple circle) will be shown on the map. After the user have key in the destination such as “TANDAS”, the destination point will pop up at the map (green rectangle). The processing will show the shortest path automatically which is the grey paths; it shows that there are only 7 steps between the starting and ending points. While on the hand, for the Figure 4.31, the destination is “SURAU” and it also take only 4 steps to reach the destination.

4.9 ZIGBEE wireless network

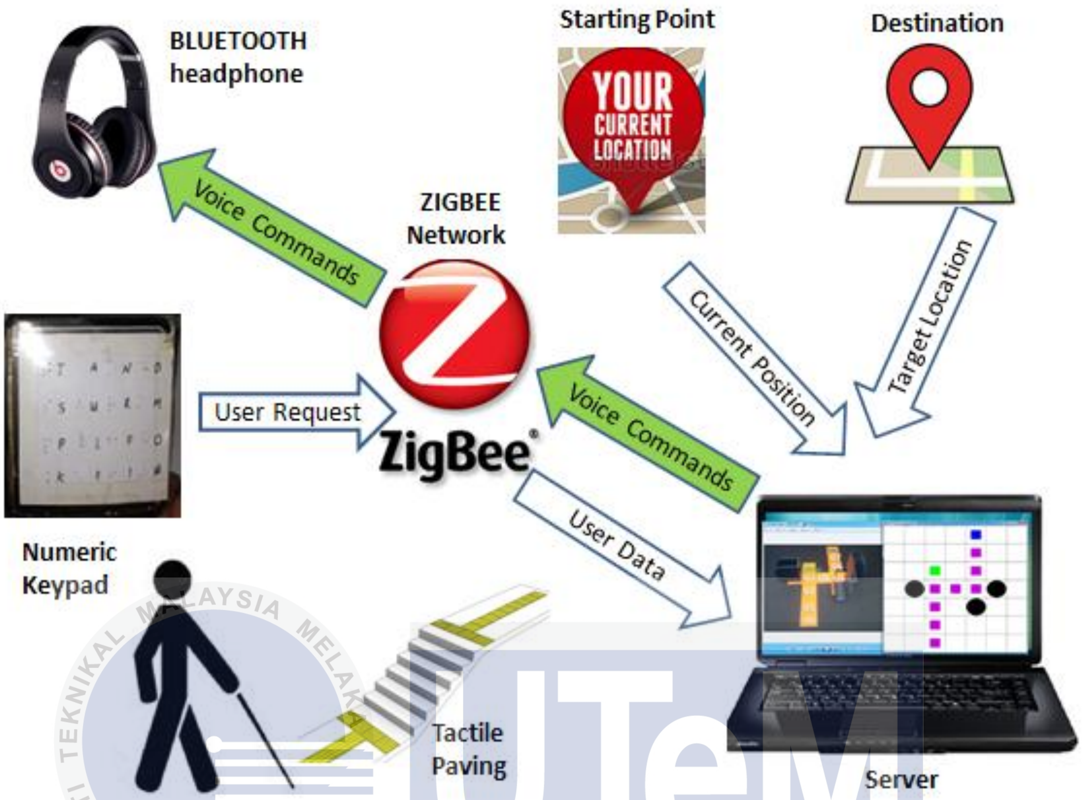


Figure 4.33 ZIGBEE wireless networks

Figure 4.33 shows the ZIGBEE wireless networks where the communication between the laptop and the blind navigation device. Blind navigation device that connected to the ZIGBEE network will estimate the user location with respect to the Processing map system. It uses the A star algorithm that is can determine the shortest path from the starting point to the end point. The orientation or direction is acquired from the digital compass which attached on the electronic cane in order to allow the system to identify the location on the map from the user current location to the target location. The ZIGBEE network acts as the center of data transferring between the navigation device and the laptop. The numeric keypad allows user to interface with the map system where it can used to key in the desired destination. The map system then identifies the address of the target. Concurrently, the RFID reader will read the tags on the tactile paving. The data of the tags of the current position and the address is sent for the map processing. Next, a set of guidance commands will be sent to the user verbally through BLUETOOTH headphone with the shortest path algorithm. The server will send the voice commands, while at the same time it will update the user position of the user. In case user takes the wrong path, the server will do path recalculation and produces the right speech again. The benefits of the system is when user need to take the corner turning, the digital compass will compare the angle and ensure the user to take the corner effectively without hitting the nearby obstacles. The server receives data from ZIGBEE network and suggested to mount at fixed locations inside the buildings. The server must be updated and the information of the destinations and objects need to be stored inside the database with respect to the map system.

4.10 A* algorithm

A* algorithm is the combination of Dijkstra's algorithm and best-first search method. The purpose of A* algorithm is to find the shortest path from start node to end node using fewer moves at the short time. It can help to generate the possibilities and picked with the least projected cost. This algorithm will divide the search area into square grid, the square grid will be determined again with walkable and unwalkable. The algorithm is performed by usually moving from the center of squares to the next center of squares. The centers of squares are called nodes.

The steps to perform the path finding are shown in Figure 4.34. Initially, begin at the start point and then adding the reachable or walkable squares into the moving list except the obstacles. Secondly, choose to travel the square in the list with the lowest score F. Path scoring F=G+H where G is movement cost from start point and H is estimated movement cost to destination. The path generation of the overall path is through by choosing the lowest score F repeatedly until reach the destination. The total number of step to complete the path is equal to the number of square moved horizontally and vertically to reach target square whereby is ignoring the obstacles and diagonal movement. It shows six steps to reach the red square.

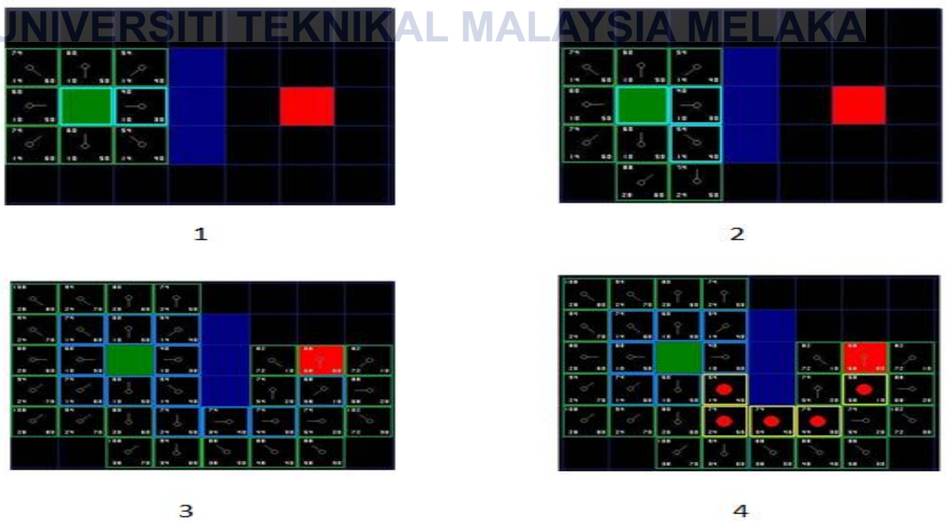


Figure 4.34: A* algorithm path finding result

4.11 Comparison between First Generation and Second Generation

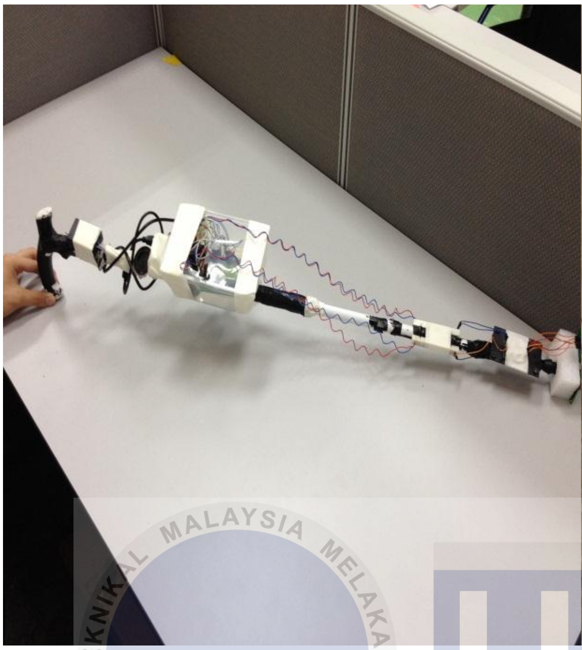


Figure 4.35: First Generation

Figure 4.36: Second Generation

The left hand side is the design of the first generation of the white cane which is shown in Figure 4.35. The components include are RFID reader, ARDUINO microcontroller, digital compass, voice module, power bank, and speaker. The function of this electronic cane is able to guide the user to travel on the tactile paving with direction guidance by voice commands through speaker. The reason is RFID tags on the tactile paving can provide the feedback information to the user about whether the user needs to go forward or turn direction or take the corner.

On the other hand, the right hand side is the design of the second generation of the electronic cane which is shown in Figure 4.36. The components include are RFID reader, ARDUINO microcontroller, digital compass, voice module, XBEE Series 2(ZIGBEE), power bank, Bluetooth module, Bluetooth headset and modified keypad with Braille Code.

The reason for upgrading to the new ones is because the shortest path algorithm can applied in the blind navigation device. Besides, additional add on features of the modified keypad with Braille code on the electronic cane made the visually impaired person easy to choose the location they want to go. The benefit of the second design performed like GPS technologies where the device allow the user navigate from location to location via point information which are stored inside the RFID tags.

There is difference between two designs. The first design focuses on travelling on the tactile paving with the feedback on tags and direction guidance from the digital compass. The second design is putting more effort on finding the shortest path throughout the travel. The A star algorithm is applied on the design. The XBEE is used to communicate between the electronic cane and the laptop. Its function is to gather to information data from the keypad which user key in the destination. The location is then sends to the laptop wirelessly for map processing. After the map system processing of finding the shortest path, the information send back to the ARDUINO and provide sound navigation through Bluetooth headset to allow user to reach destination.

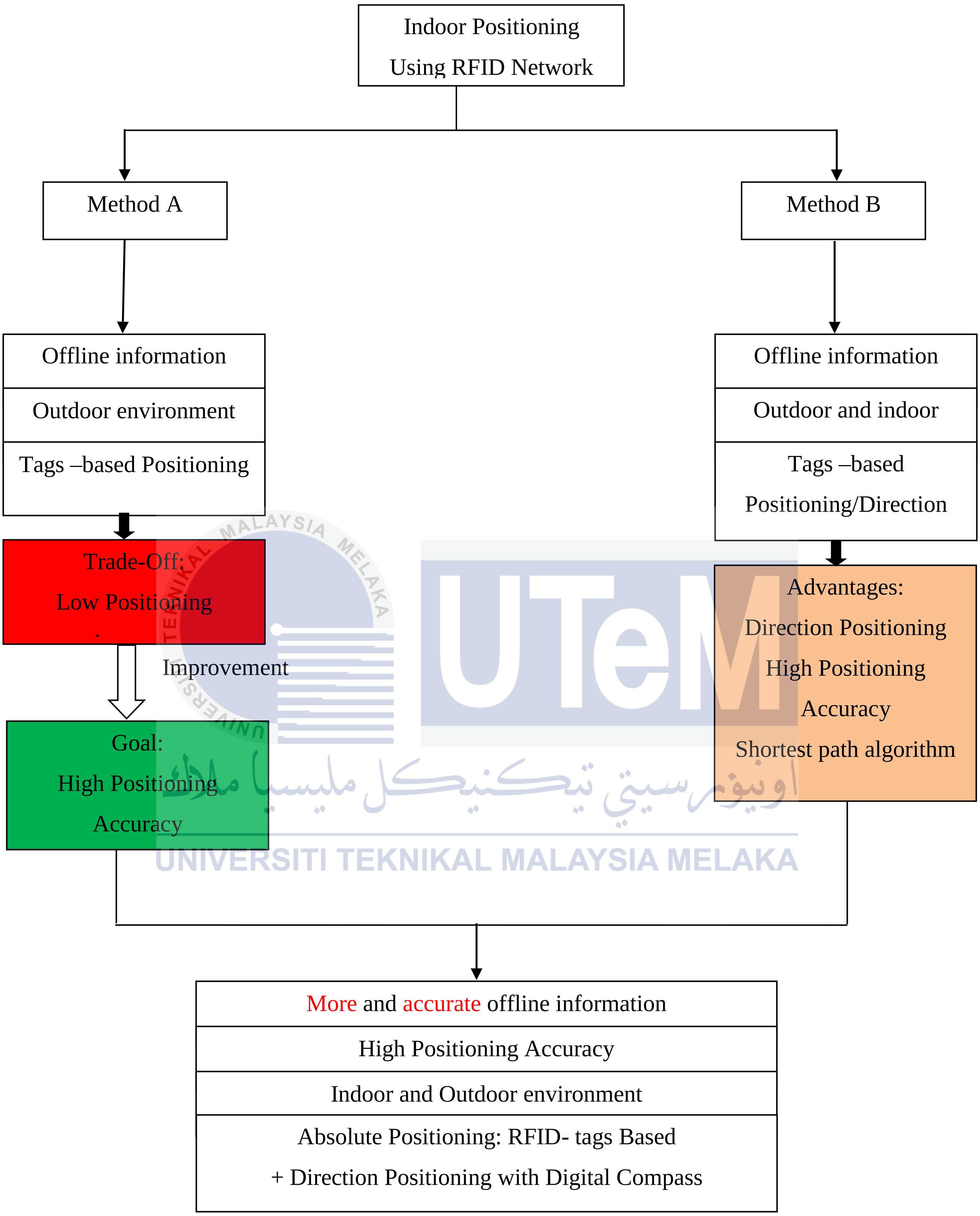


Figure 4.37:Trade-off between Method A and B

Weakness
Improvement
Advantages

4.12 off between method of indoor positioning using RFID network

Table 4.8: Indoor Positioning Using RFID Network

	Method A	Method B
Way for indoor positioning	Offline information collection	Offline information collection
Location	Outdoor environment	Outdoor and indoor environment
Method	Tags-based Positioning	Tags-based Positioning/ Direction Guidance
Strength	-	High Positioning Accuracy with Direction Guidance Shortest path algorithm
Weakness	Low Positioning Accuracy	-
Improvement Goals	High Positioning Accuracy	-

There is a comparison between Method A and Method B which refer to Figure 4.37. Both methods are implemented for the indoor positioning using RFID Network. The only difference is method A is only indoor navigation by reading the RFID tags which is the conventional method while the method B is indoor positioning with RFID network plus direction navigation using digital compass module with the shortest path algorithm.

Method B is chosen because of high accuracy positioning and direction pointing. This method can read the RFID tags on the footpath and guide the user to travel through the navigation sound from speaker or headset. Besides, the digital compass module is always activated to measure the orientation angle of the user. If the user is travelling out of the path, the navigation device will warn the user and let him/her knows immediately. In addition, the shortest path algorithm is benefit for visual impaired people as they can saves times on travelling.

Due to safety reasons, method B has more advantages than method A because this method can be used only in outdoor but also at indoor environment. Furthermore, this method B is proven to be safe as the obstacles avoidance is being tested by applying the digital compass on the map system. On the other hand, method A is applied at outdoor environment only. Method A requires GPS (Global Positioning System) to travel at indoor however it is not functional at indoor environment due to signal disturbance.

In order to achieve the objective of indoor navigation for the visually impaired people, method A and method B can be combined and create a new concept to achieve high accuracy positioning which is applicable at indoor environment.

4.13 Summary of Results and Discussion

This chapter explains the results of the experiment 1, 2 and 3. The first experiment is about to test the accuracy of digital compass where the compass is used to guide the heading direction which lead to the path of destination. The second experiment is to test the validity of A* algorithm before implementation on the second blind navigation device. The third experiment is to evaluate the performance of the second blind navigation device in terms of lab test and validity of the algorithm.

For the case of Experiment 1, the digital compass is able to produce the high accuracy compass reading for the orientation heading to the destination. The participant for the experiment is able to go straight, turn left and turn right under the voice commands and reach the destination easily.

For the case of Experiment 2, the testing of the A star algorithm on the mobile robot shows that the mobile robot is able to take corner effectively and able to avoid the obstacles. The time needed for mobile robot to reach the destination is consistent and justifiability, however the time needed for the visual impaired person is very subjective and dependent on the person himself where the time taken is longer.

For the case of Experiment 3, the evaluation of the performance of the second blind navigation device shows the participants with eye-closed are able to reach the destination with the voice commands from the navigation device. The map system is able to synchronize with RFID technology through ZIGBEE network to provide the correct path guidance with the shortest path to reach the destination.

CHAPTER 5

CONCLUSION

This chapter will conclude the research finding and the evaluation performance of the navigation device using RFID network for visually impaired people. Recommendations for the future work will be discussed.

5.1 Conclusion

The navigation device developed using RFID network is expected to guide proper walking for visually impaired person through compass. The blind navigation device using RFID technology with digital compass and voice system is great at travelling at the right track with correct direction. The implementation of A* algorithm in blind navigation device will help to determine the shortest path saves lots of time and energy on travelling. The performances proved that the device designed is robustness and the result is valid and reliable.

5.2 Future Work

Overall, the navigation device is constructed as planned. Referring to the design concept and components selection, the design of this navigation device might have some improvement to make it better such as considering the product weight and sustainability design. Advanced technology could be implemented in the navigation design.

These suggestions are as below;

a) Indoor Wi-Fi Positioning System

Wi-Fi based positioning system (WPS) is implemented where global positioning system (GPS) is disallowed at indoor due to some causes including multipath and signal blockage. As the wireless technology becomes more advanced, the device that the wireless network can used to locate the objects as well as people. This localization technique for indoor positioning with multiple wireless access point will increase the signal strength to the receiver. With the information stored in the database, user can access the data easily to navigate for the destination.

b) Indoor Navigation Using Smart Phone

Smart Phone becomes the useful communication tools nowadays and everyone afford to have at least one mobile phone either Android phone or IPhone. Software applications can be created and implemented inside the smart phone for indoor navigation. The built-in sensor like compass, gyro sensor and magneto sensor enable to give the direction navigation. Various navigation applications can be created to help the visually impaired person to travel independently.

c) Visible Light Communication

Indoor navigation system use visible light communication where use LED lights and geomagnetic correction method to guide the visually impaired people to travel. It becomes easier to locate the user’s position within a range of 1-2m. Besides the system can determine which floor that they are located on.

REFERENCE

[1] WHO secretariat, "WHO | Visual Impairment and Blindness." Available: http://www.who.int/mediacentre/factsheets/fs282/en/ [Accessed: June 2012].

[2] U. Biader Ceipidor, C.M. Medaglia, F.Rizzo, A.Serbanati, "Radio Virgilio/Sesamonet: an RFID-based Navigation system for visually impaired"pp.1-6

[3] P. Dharani, "RFID Navigation System for the Visually Impaired," 2012.

[4] S. Willis and S. Helal, "RFID information grid for blind navigation and wayfinding," in *Wearable Computers, 2005. Proceedings. Ninth IEEE International Symposium on*, 2005, pp. 34–37.

[5] Y. Shiizu, Y. Hirahara, K. Yanashima, and K. Magatani, "The development of a white cane which navigates the visually impaired," in *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, 2007, pp. 5005–5008.

[6] A. Y. Szeto and S. K. Sharma, "RFID based indoor navigational aid for persons with severe visual impairments," in *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, 2007, pp. 6360–6363.

[7] E. Wise et al., "Indoor Navigation for the Blind and Vision Impaired: Where are we and where are we going?," in *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*, 2012, pp. 1–7.

[8] S. Mau, N. A. Melchior, M. Makatchev, and A. Steinfeld, "BlindAid: An ElectronicTravel Aid for the Blind," *Robotics Institute Report, CMU-RI-TR-07-39*, 2008.

[9] M. Jain, R. C. Rahul, and S. Tolety, "A study on Indoor navigation techniques using smartphones," in *Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on*, 2013, pp. 1113–1118.

[10] M. Sarfraz and S. A. J. Rizvi, "Indoor Navigational Aid System for the Visually Impaired," in *Geometric Modeling and Imaging, 2007. GMAI'07*, 2007, pp. 127–132.

[11] S. S. Santhosh, T. Sasiprabha, and R. Jeberson, "BLI-NAV embedded navigation system for blind people," in *Recent Advances in Space Technology Services and Climate Change (RSTSCC), 2010*, 2010, pp. 277–282.

[12] M. H. Choudhury, D. Aguerrevere, and A. B. Barreto, "A Pocket-PC Based Navigational Aid for Blind Individuals," in *Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2004.(VECIMS). 2004 IEEE Symposium on*, 2004, pp. 43–48.

[13] A. J. Fukasawa and K. Magatani, "A navigation system for the visually impaired an intelligent white cane," in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, 2012, pp. 4760–4763.

[14] J. Sánchez, "Mobile audio navigation interfaces for the blind," in *Universal Access in Human-Computer Interaction. Intelligent and Ubiquitous Interaction Environments*, Springer, 2009, pp. 402–411.

[15] M. Bousbia-Salah, A. Redjati, M. Fezari, and M. Bettayeb, "An ultrasonic navigation system for blind people," in *Signal Processing and Communications, 2007. ICSPC 2007. IEEE International Conference on*, 2007, pp. 1003–1006.

[16] E. B. Kaiser and M. Lawo, "Wearable navigation system for the visually impaired and blind people," in *Computer and Information Science (ICIS), 2012 IEEE/ACIS 11th International Conference on*, 2012, pp. 230–233.

[17] M. Shamsi, M. Al-Qutayri, and J. Jeedella, "Blind assistant navigation system," in *Biomedical Engineering (MECBME), 2011 1st Middle East Conference on*, 2011, pp. 163–166.

[18] S. Chumkamon, P. Tuvaphanthaphiphat, and P. Keeratiwintakorn, "A blind navigation system using RFID for indoor environments," in *Electrical*

Engineering/Electronics, Computer, Telecommunications and Information Technology, 2008. ECTI-CON 2008. 5th International Conference on, vol. 2, 2008, pp. 765–768.

[19] A. Ganz, S. R. Gandhi, C. Wilson, and G. Mullett, "INSIGHT: RFID and Bluetooth enabled automated space for the blind and visually impaired," in *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, 2010, pp. 331–334.

[20] R. K. Megalingam, A. P. Rajendran, and D. Dileepkumar, "LARN: Implementation of automatic navigation in indoor navigation for physically challenged," in *India Conference (INDICON), 2012 Annual IEEE*, 2012, pp. 956–961.

[21] K.-C. Fan and P.-C. Lui, "Solving find path problem in mapped environments using modified A* algorithm," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 24, no. 9, pp. 1390–1396, 1994.

[22] N. A. Vlassis, N. M. Sgouros, G. Efthivoulidis, G. Papakonstantinou, and P. Tsanakas, "Global path planning for autonomous qualitative navigation," in *Tools with Artificial Intelligence, 1996., Proceedings Eighth IEEE International Conference on*, 1996, pp. 354–359.

[23] G. E. Jan, M. C. Lee, S. Hsieh, and Y.-Y. Chen, "Transportation network navigation with turn penalties," in *Advanced Intelligent Mechatronics, 2009. AIM 2009. IEEE/ASME International Conference on*, 2009, pp. 1224–1229.

[24] Y. Toda and N. Kubota, "Path planning using multi-resolution map for a mobile robot," in *SICE Annual Conference (SICE), 2011 Proceedings of*, 2011, pp. 1276–1281.

[25] F. Xiao-wei, L. Zhong, and G. Xiao-guang, "Path planning for UAV in radar network area," in *Intelligent Systems (GCIS), 2010 Second WRI Global Congress on*, vol. 3, 2010, pp. 260-263.

[26] C. Liu and J. Wu, “Swing: Small world iterative navigation greedy routing protocol in manets,” in *Computer Communications and Networks, 2006. ICCCN 2006. Proceedings. 15th International Conference on*, 2006, pp. 339–350.

APPENDIX A

Final Year Project Timeline

	Schedule									
	2013					2014				
Project Activities	Sept	Oct	Nov	Dec	Jan	Feb	Mac	Apr	May	Jun
Propose Project Title										
Literature Review										
Methodology Proposal										
Stage1:First Design										
Stage2:Hardware Assembly 1										
Stage3:Testing and Troubleshooting										
Stage4:Experiment 1										
Data Collection & Analysis (1st Design)										
Review & Improvement										
Stage5:Second Design										
Stage6:Hardware Assembly 2										
Stage7:Testing and Troubleshooting										
Stage8:Experiment 2 & 3										
Data collection & analysis(2nd Design)										
Performance Evaluation(2nd Design)										
Report Writing										
Presentation and Demonstration										

APPENDIX B

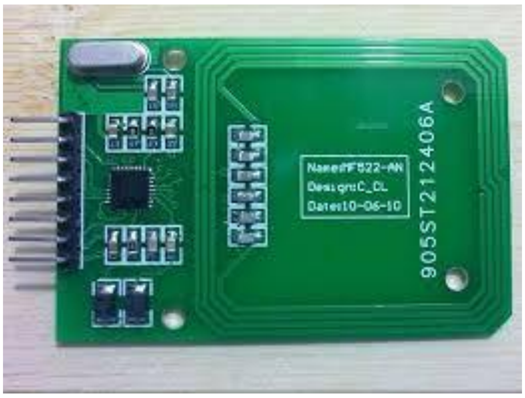
ARDUINO Pro Mini Datasheet

ARDUINO Pro Mini On-board Microcontroller, Atmega 328, 16MHz

Microcontroller	ATmega168
Operating Voltage	3.3V or 5V (depending on model)
Input Voltage	3.35 -12 V (3.3V model) or 5 - 12 V (5V model)
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	8
DC Current per I/O Pin	40 mA
Flash Memory	16 KB (of which 2 KB used by boot loader)
SRAM	1 KB
EEPROM	512 bytes
Clock Speed	8 MHz (3.3V model) or 16 MHz (5V model)

APPENDIX C

MIFARE High Frequency RFID Module



- Complete Read/Write module
- Auto checks for presence of a tag
- Contactless operating frequency13.65MHz
- TTL RS232 Interface, baud rate 19200bps
- Fast data transfer Contactless communication up to 106KHz
- Secure Encrypted contactless communication
- Ideal for money, secure access and fast data applications
- Operating voltage: DC 3.0-5.5V
- Unique serial number on each device
- Size; 40mm x 25mm x 6mm

APPENDIX D

Digital compass HMC6352

- Simple I2C interface
- Voltage Range (2.7V-5.5V)
- 0.5 degree heading resolution
- 1 degree repeatability

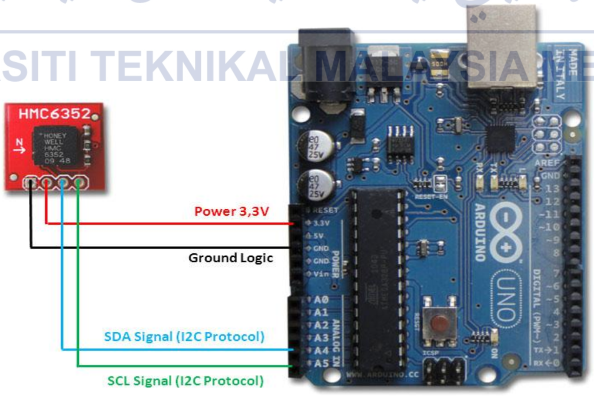
Label:

GND – G round

VCC – 5V power input

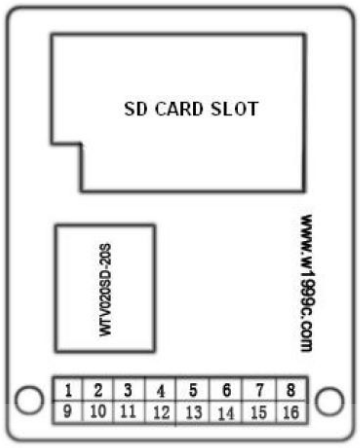
SDA – Serial Data Line

SCL –Serial Clock Line



APPENDIX E

Voice Module WTV020-SD



Voice Module Data Sheet

PIN	SYS.	Function	PIN	SYS.	Function
1	DC +3.3V	+3.3V	9	GND	GND
2	SPK+	Audio Output	10	DC +3.3V	+3.3 V
3	P07	I/O	11	SPK+	Audio output
4	P03	I/O	12	SPK-	Audio output
5	NC	NC	13	P06	BUSY
6	NC	NC	14	RST	Reset
7	P02	I/O	15	P04	I/O
8	NC	NC	16	P05	I/O

APPENDIX F

Slave RFID

/* Initialize header for Software Serial, WTV, RFID */

```
#include <stdio.h>
#include <Wire.h>
#include <SoftwareSerial.h>  // Software Serial Header
#include <Wtv020sd16p.h>  // Wtv sound module header
char c;
int goal_Tx = 0;
int goal_Ty = 0;
int goal_Sx = 0;
int goal_Sy = 0;

***************************************************************************
***
```


/*Initialize pins for Wtv Sound Module */

```
int resetPin = 7; // Reset pin from Wtv to pin 6 Arduino
int clockPin = 8; // Clock pin from Wtv to pin 7 Arduino
int dataPin = 12; // Data pin from Wtv to pin 8 Arduino
int busyPin = 13; // Busy pin from Wtv to pin 9 Arduino
Wtv020sd16p wtv(resetPin, clockPin, dataPin, busyPin);

***************************************************************************
***
```

/* Initialize pins for RFID Reader */

```
SoftwareSerial rfSerial(2,3); // (RX,TX) -> RFID RX to pin 2 & TX at pin 3
byte data[20]; // Stores 20 X 8-bit unsigned number from RFID
int val = 0; // Convert analog readings from RFID to Digital
int tag_scan = 0; // Extract 1-bit data from RFID tag
int tag_scan2 = 0;
int tag_memory = 0;
int tag_memory2 = 0;

***************************************************************************
***
```

/* Initialize pins for Compass Module HMC6352 */

```
int HMC6352Address = 0x42;
// This is calculated in the setup() function
int slaveAddress;
boolean ledState = false;
byte headingData[2];
int i, headingValue, Compass_val;

***************************************************************************
***
```


/* Initialize variables for User Position */

```
int user_x; // User position on coordinate-x
int user_y; // User position on coordinate-y
int goal_x = 0; // Goal position on coordinate-x
int goal_y = 0; // Goal position on coordinate-y

***************************************************************************
***
```

/*Declaration for Start, goal and obstacle with binary */

```
int nothing = 0; // walkable road
int wall = 255; // obstacle
int user = 254; // user location
int goal = 1; // goal location

*****************************************************************************
***

/*Set Starting Map Location on (0,0) */

int x = 0; // Map starting point
int y = 0; // Map starting point

*****************************************************************************
***

/*Initialize temporarily Variables */

int temp_A = 0; // temporarily location
int temp_B = 0;
int counter = 0; // count up value

*****************************************************************************
***

/*Initialize variables for node searching on Map drawn */

int minimum_node = 250; // minimum node checked
int min_node_location = 250; // minimum node position located
int new_state = 1;
int old_state = 1;
int trans = 50;
int reset_min = 250;
```

```
*****************************************************************************
***
```

/* Map Drawn for the Path Finding */

```
int map1[10][10] = {{0,0,0,0,0,0,0,0,0,0},
                    {0,0,0,0,0,0,0,0,0,0},
                    {0,0,0,0,0,0,0,0,0,0},
                    {0,0,255,255,0,0,0,0,0,0},
                    {0,0,0,0,0,0,0,0,0,0},
                    {0,0,0,0,0,0,0,0,0,0},
                    {0,0,0,0,255,0,0,255,0,0},
                    {0,0,0,0,255,0,0,255,0,0},
                    {0,0,0,0,255,0,0,0,0,0},
                    {0,0,0,0,0,0,0,0,0,0}};

*****************************************************************************
***
```


/*Initialize Setup */

```
void setup()
{
  slaveAddress = HMC6352Address >> 1;   // This results in 0x21 as the address to pass to
TWI
  Wire.begin();       // join i2c bus (address optional for master)
  Serial.begin(9600);  // start serial for output
  rfSerial.begin(19200); // Set RFID software serial port as 19200 baud rate
  wtv.reset();
*****************************************************************************
***
```

/* Read RFID Tag */

```
void readData(){
  int val = rfSerial.read();  // Read serial data and convert to int
  // While tag detected is not programmed to be start with header 0xAA, continue scanning
  while(val != 0xAA){
    val = rfSerial.read();  // Read tag if correct tag detected
    delay(10);  // delay program for 10ms
  }

  rfSerial.read();  // Read 1st byte -> 0xAA
  rfSerial.read();  // Read 2nd byte -> 0xBB
  rfSerial.read();  // Read 3rd byte -> 0x0A (Block Read)
  data[0] = rfSerial.read();
  data[1] = rfSerial.read();
  data[2] = rfSerial.read();
  data[3] = rfSerial.read();
  data[4] = rfSerial.read();
  data[5] = rfSerial.read();
  data[6] = rfSerial.read();
  data[7] = rfSerial.read();
  data[8] = rfSerial.read();
  data[9] = rfSerial.read();
  data[10] = rfSerial.read();
  data[11] = rfSerial.read();
  data[12] = rfSerial.read();
  data[13] = rfSerial.read();
  data[14] = rfSerial.read();
  data[15] = rfSerial.read();
  tag_scan = data[15];  // Insert byte data[15] into tag_scan
  Serial.print(tag_scan);
  Serial.print(" ");
  Serial.println();
}
```

```
**************************************************************************
***
```

/* Protocol for RFID Reader */

```
 void cardData(){
   rfSerial.write(0xAA);  // This 1 byte header should always be 0xAA
   rfSerial.write(0xBB);  // This 1 byte header should always be 0xBB
   rfSerial.write(0x0A);  // For block read purpose, this is set as 0x0A
   rfSerial.write(0x21);  // This byte command module to perform block read
   rfSerial.write((byte)0x00);  // This byte set as 0x00 for key A or 0x01 for key B
   rfSerial.write(0x04);  // Block number selected when program RFID tag
   rfSerial.write(0xFF);  // For Block read, this is set as FFFFFFFFFFFF for keys
   rfSerial.write(0xFF);
   rfSerial.write(0xFF);
   rfSerial.write(0xFF);
   rfSerial.write(0xFF);
   rfSerial.write(0xFF);
   rfSerial.write(0x2F);   // check sum is XOR all bytes in the packet except header and
CSUM byte
   rfSerial.println();  // Print next line on Monitor
   readData();  // Call function readData();
  }

**************************************************************************
***
```

/* Function for Compass*/

```
void compass_hmc()
```

```
{
 Wire.beginTransmission(slaveAddress);
 Wire.write("A");              // The "Get Data" command
 Wire.endTransmission();
 delay(10);                   // The HMC6352 needs at least a 70us (microsecond) delay
 Wire.requestFrom(slaveAddress, 2);       // Request the 2 byte heading (MSB comes first)
 i = 0;
 while(Wire.available() && i < 2)
 {
  headingData[i] = Wire.read();
  i++;
 }
 headingValue = headingData[0]*256 + headingData[1];   // Put the MSB and LSB
together
 Compass_val = int(headingValue / 10);
//  Serial.println("Compass Value=");    //open back to get the compass value
//  Serial.print(Compass_val);
 delay(100);
}

***************************************************************************
***

void receiveEvent(int howMany)
{
 while(Wire.available())    // slave may send less than requested
 {
  c = Wire.read(); // receive a byte as character
  if(c=='T')
  {
   wtv.asyncPlayVoice(1);
   delay(500);
  }
  if(c=='A')
```

```
{
  wtv.asyncPlayVoice(2);
  delay(500);
}
if(c=='N')
{
  wtv.asyncPlayVoice(3);
  delay(500);
}
if(c=='D')
{
  wtv.asyncPlayVoice(4);
  delay(500);
}
if(c=='S')
{
  wtv.asyncPlayVoice(5);
  delay(500);
}
if(c=='U')
{
  wtv.asyncPlayVoice(6);
  delay(500);
}
if(c=='R')
{
  wtv.asyncPlayVoice(7);
  delay(500);
}
if(c=='M')
{
  wtv.asyncPlayVoice(8);
  delay(500);
}
```

```
if(c=='P')
{
  wtv.asyncPlayVoice(9);
  delay(500);
}
if(c=='L')
{
  wtv.asyncPlayVoice(10);
  delay(500);
}
if(c=='F')
{
  wtv.asyncPlayVoice(11);
  delay(500);
}
if(c=='O')
{
  wtv.asyncPlayVoice(12);
  delay(500);
}
if(c=='K')
{
  wtv.asyncPlayVoice(13);
  delay(500);
}
if(c=='E')
{
  wtv.asyncPlayVoice(14);
  delay(500);
}
if(c=='I')
{
  wtv.asyncPlayVoice(15);
  delay(500);
```

```
    }
    if(c=='#')
    {
      wtv.asyncPlayVoice(16);
      delay(500);
    }
  }
}
***************************************************************************
***
```

/* Function for DFS1 */

```
void DFS_1(int user_x, int user_y)
{
  for(x=0; x<10; x++){
    for(y=0; y<10; y++)
      if(map1[x][y] != wall && map1[x][y] != goal)
      map1[x][y] = nothing;
  }

    map1[user_x][user_y] = user;
```

/* Declare int function for minimum surrounding node to pass value back to DFS2 */

```
int min_surrounding_node_value(int x, int y)
{
  minimum_node = reset_min;
```

```
 if(x<9)
        if(map1[x+1][y] < minimum_node && map1[x+1][y] != nothing)
        {
          minimum_node = map1[x+1][y];
          min_node_location = 3;
        }

 if(x>0)
        if(map1[x-1][y] < minimum_node && map1[x-1][y] != nothing)
        {
          minimum_node = map1[x-1][y];
          min_node_location = 1;
        }

 if(y<9)
        if(map1[x][y+1] < minimum_node && map1[x][y+1] != nothing)
        {
          minimum_node = map1[x][y+1];
          min_node_location = 2;
        }

 if(y>0)
        if(map1[x][y-1] < minimum_node && map1[x][y-1] != nothing)
        {
          minimum_node = map1[x][y-1];
          min_node_location = 4;
        }
 return minimum_node;
}

******************************************************************************
***
```


/* Declare function for DFS2 */

```
int DFS_2(int user_x, int user_y)
{
  DFS_1(user_x, user_y);

  Up1:
  while(c!='1' && c != '2')
  {
  // Check goal selected by user and add into Map
    //Serial.println("Waiting for command...");
    delay(300);
  }
if(c=='1') //tandas
  {
    wtv.asyncPlayVoice(18);
    delay(1500);
    goal_x = 6;
     goal_y = 2;
     map1[goal_x][goal_y] = goal;
     delay(10);
  }
  else if(c=='2')// surau
  {
    wtv.asyncPlayVoice(19);
    delay(1500);
    goal_x = 4;
   goal_y = 0;

    map1[goal_x][goal_y] = goal;  // Set goal coordinate
   delay(10);
  }
  else if(c=='4')
  {
    wtv.asyncPlayVoice(26);
```

```
  delay(100);
  goto Up1;
}

counter = 0;

while(counter < 50)
{
  x = 0;
  y = 0;
  while(x<10 && y<10)
  {
    if(map1[x][y] != wall && map1[x][y] != goal)
    {
      if(min_surrounding_node_value(x,y) < reset_min && map1[x][y] == user)
      {
        return min_node_location;
      }

      else if(minimum_node != reset_min)
      {
        map1[x][y] = minimum_node+1;
      }
    }

      y++;
      if(y==10 && x!=10)
      {
        x++;
        y=0;
      }
    }
counter++;
    }
```

```
    return 0;
}

*****************************************************************************
***

/* Declare function for Finale */

int Final_DFS()
{
  compass_hmc();
  while(map1[user_x][user_y] != goal)

  {
    new_state = DFS_2(user_x, user_y);

    /* Forward */
    if(new_state==3)
    {
      while(Compass_val<30 && Compass_val>0 || Compass_val>350 &&
Compass_val<359)
      {
        compass_hmc();
      }
      user_x++;
      Listen1:
      cardData();
      wtv.asyncPlayVoice(21);  // Play sound module "Forward"
      delay(2000);
      if(data[15]==255 || (data[14]==user_y && data[15] == user_x+1))
      {
        goto Listen1;
      }
      delay(2000);
```

```
}

/* Right */
if(new_state==2)
{
 while(Compass_val<80 && Compass_val>130)
 {
  compass_hmc();
 }
 user_y++;
 Listen2:
 cardData();
 wtv.asyncPlayVoice(20); // Play sound module "Right"
 delay(2000);
 if(data[15] == 255 || (data[14] == user_y-1 && data[15] == user_x))
 {
  goto Listen2;
 }
 wtv.asyncPlayVoice(21);  // Play sound module "Forward"
 delay(2000);
 delay(2000);
}

/* Reverse */
if(new_state==1)
{

 while(Compass_val<230 && Compass_val>180)
 {
  compass_hmc();
 }
 user_x--;
 Listen3:
 cardData();
```

```
    wtv.asyncPlayVoice(24);   // Play sound module "Reverse direction Please Turn 180 degree and walk forward"
    delay(2000);
    if(data[15] == 255 || (data[14] == user_y && data[15] == user_x-1))
    {
      goto Listen3;
    }
    delay(2000);
  }

  /* Left */
  if(new_state==4)
  {
    while(Compass_val>290 && Compass_val<340)
    {
      compass_hmc();
    }
    user_y--;
    Listen4:
    cardData();
    wtv.asyncPlayVoice(22);  // Play sound module "left"
    delay(2000);
    if(data[15] == 255 || (data[14] == user_y+1 && data[15] == user_x))
    {
      goto Listen4;
    }
    delay(2000);
  }

  old_state = new_state;
  trans--;
  }
  wtv.asyncPlayVoice(17);
  delay(2000);
```

```
  return 0;
}

************************************************************************
***
/* Function to reset all variables after path found */

void val_res()
 {
  goal_x = 0;
  goal_y = 0;
  DFS_1(user_x,user_y);
 }

************************************************************************
***

/* Declared void loop function */

void loop()
{
  Wire.begin(5);
  Wire.onReceive(receiveEvent);
  wtv.playVoice(0);
  delay(2000);
  Scan_Back:
  cardData();
    while(tag_scan==255)
  {
    goto Scan_Back;
  }
  tag_memory = tag_scan;
  tag_memory2 = tag_scan2;
  user_x = tag_memory;
```

```
  user_y = tag_memory2;
  Final_DFS();
  c = 0;
  delay(100);
  val_res();
  delay(100);
}
```

APPENDIX G

Master Keypad

```
#include <Wire.h>
#include <Keypad.h>

const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns
int i = 0;

char hexaKeys[ROWS][COLS] = {
  {'T','A','N','D'},
  {'S','U','R','M'},
  {'P','L','F','O'},
  {'K','E','I','#'}
};

byte rowPins[ROWS] = {5, 4, 3, 2}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {9, 8, 7, 6}; //connect to the column pinouts of the keypad
Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);

char PinT[6] = {'T','A','N','D','A','S'};
char PinT1[6] = {0,0,0,0,0,0};
char Pin_Surau[5] = {'S','U','R','A','U'};
char Pin_Surau1[5] = {0,0,0,0,0};
char PinA[3] = {'A','T','M'};
char PinA1[3] = {0,0,0};
int z = 0;
int z1=0;
int z2 = 0;
```

```
void setup()
{
  Wire.begin();
  Serial.begin(9600);
}

void checkPin()
{
  int correct = 0;
  int correct1=0;
  int correct2=0;
  int i,j,k;
  for(i=0;i<6;i++)
  {
    if(PinT1[i]==PinT[i])
    {
      correct++;
    }  }
      for(k=0;k<5;k++)
  {
      if(Pin_Surau1[k]==Pin_Surau[k])
    {
      correct2++;
    }
  }

  for(j=0;j<3;j++)
  {
      if(PinA1[j]==PinA[j])
    {
      correct1++;
    }
  }
```

```
if(correct==6)
{
   Wire.write('1');
   Serial.println("TANDAS");
delay(500);
}
  else if(correct2==5)
{
    Wire.write('2');
   Serial.println("SURAU");
delay(500);
}
else if(correct1==3)
{
    Wire.write('3');
   Serial.println("ATM");
delay(500);
}

else
{
  Wire.write('4');
 Serial.println(4);
delay(500);
}
for (int zz=0; zz<6; zz++)
{
  PinT1[zz]=0;
}


  for (int zzzz=0; zzzz<5; zzzz++)
{
  Pin_Surau1[zzzz]=0;
```

```
 }
  for (int zzz=0; zzz<3; zzz++)
 {
  PinA1[zzz]=0;
 }
}

void loop()
{
  Wire.beginTransmission(5);
  char customKey = customKeypad.getKey();

   if (customKey!=NO_KEY){
  Wire.write(customKey);
  //Serial.print(customKey);
   switch(customKey)
   {
    case '#':
    z=0;
    z2=0;
    z1=0;
    delay(100);
    checkPin();
    break;
    default:
    PinT1[z]=customKey;
    Pin_Surau1[z1]=customKey;
    PinA1[z1]=customKey;

    z++;
    z2++;
    z1++;
   }
 }
```

```
  Wire.endTransmission();
  delay(10);
}
```

APPENDIX H

XBEE Connection

```
void setup()
{
Serial.begin(9600);
}

void loop() {
  while(Serial.available() > 0)
{
  Serial.write(Serial.read());
  delay(10);
  }



}
```