Letter of Transmittal

November 6th, 2014

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser University
Burnaby, British Columbia
V5A 1S6

Re: ENSC 440 and 305W Functional Specification for the VIA: Visually Impaired Assistant

Dear Dr. Rawicz,

You will find enclosed the design specifications for the VIA: Visually Impaired Assistant. Our hope is to develop a remote-like device that will allow the visually impaired to navigate easily in any environment. Our device will allow any client that uses our product to feel they can live a more independent life as they receive more knowledge about their surroundings.

This purpose of this design specification is to provide a comprehensive technical overview of the VIA. It is meant to detail our multiple step design process, the system specifications, the in-depth design of our system, as well as an extensive test plan to ensure correct functionality of the system.

Sensible Solutions is composed of three engineering students: Ahmad Ibrahim, Robert Sanchez, and Jessica Zanewich. Our company is dedicated and focussed on the VIA project. We appreciate you taking the time to look over our design specification. If there are any questions or concerns regarding this document, please feel free to contact us by email at jzanewic@sfu.ca.

Sincerely,

Jessica Zanewich
Chief Executive Officer

Enclosed: Design Specification for the Visually Impaired Assistant (VIA)

Sensible Solutions

Design Specification for the

VIA: Visually Impaired Assistant

Project Team: Ahmad Ibrahim
Jessica Zanewich
Robert Sanchez

Contact Person: Jessica Zanewich

Submitted to: Dr. Andrew Rawicz – ENSC 440
Steve Whitmore – ENSC 305

School of Engineering Science
Simon Fraser University

Issued Date: November 6th, 2014

Revision: 1.1

Sensible Solutions

Abstract

The goal of the VIA is to create a safe and easily traversed setting for any visually impaired person. As a company, we must think of the most important aspects of being visually impaired in today's society. This means it is very important to think of specific aspects when designing the device; this includes, but is not limited to, convenience, portability, and environment. Our concept for the VIA takes all of these components into account. This will be seen in the following documentation.

This design specification is meant to provide an in-depth discussion of the main technical components of our VIA system. Included are the electrical, embedded and software components involved in the proof-of-concept model outlined in the Functional Specifications. Our intention is to create a small device that will use audio feedback along with ultrasonic sensors to provide the necessary information for the user to navigate safely.

You will find an outline of the process details (as initially proposed in the Functional Specifications), the electronics design (including hardware and software), and a test plan. This test plan will test the validity of our proof-of-concept and will be used throughout the design process to improve on that design.

Sensible Solutions

Table of Contents

List of Figures

Glossary

Accelerometer – Measures acceleration

Adafruit Trinket Pro – Referred to as Trinket or Trinket Pro; The microcontroller used in the marketable, and final product

Arduino Uno – Referred to as Arduino; The microcontroller used during prototyping stages

Braille – Written language used by the visually impaired; represented by patterns of raised dots

Embedded System – A computer system with a dedicated function

Gyroscope – Determines orientation with respect to the force of gravity

Microcontroller – A small computer designed to govern embedded systems operations

MM – Millimeter(s)

MS – Millisecond(s)

PCB – Printed circuit board

Proof of concept – a demonstration in principle, whose purpose is to verify that some concept or theory has the potential of being used

S – Second(s)

Ultrasonic Sensor – Sensors working off the principle of radar, or sonar

μS – Microsecond(s)

V – Voltage

VIA – Visually Impaired Assistant

Watchdog timer – An electronic timer used to detect, and recover from computer malfunctions

Sensible Solutions

1. Introduction

The Visually Impaired Assistant, or the VIA, is a handheld device created by Sensible Solutions. The product uses ultrasonic sensors to detect and identify nearby objects, as well as give an approximation of the distance the obstacle is from the user through a small speaker. It is a more expensive option than the currently used white cane, but with much more information provided to the user to make navigation easier. The system must meet the requirements of the user, as well as the standards set in place for the different components used. These requirements will be outlined in the following documentation.

1.1 Scope

The design specification presented today is meant to support the Functional Specification previously provided, as well as enhance knowledge of the technical specifications of the VIA project. The technical specifics of the product will be outlined throughout this document, which includes the Arduino Uno, Adafruit trinket pro, ultrasonic sensors, audio feedback system, accelerometer and gyroscope, and the software used to connect all components.

As well, the design specification will give you great insight into the following:
- Discussion of the system specifications in great detail
- An in-depth review of the design process steps
- A more specific outline of tests for the technical components

1.2 Intended Audience

The design specification will be used by many people over the course of the development of the VIA. Each member of Sensible Solutions will use this document for the project. Our Chief Technical Officer, Rob Sanchez, will also use this to guide the technical design of the system. Any future developers on the project will also be able to use this document as a useful tool for implementation of the VIA. Finally, we hope any investors and stakeholders can get a greater insight into the VIA project through the design specification.

1.3 Background

The VIA is an extremely important technology that could greatly improve the lives of the visually impaired. It is designed to be portable, allow for a greater sense of surroundings, and to provide a great amount of information to the user. Rather than using the white cane currently in circulation, the user will get a better sense of the specific object and distance it is from the person, to create a more fluid experience in everyday life.

To be considered legally blind, central vision must be 20/200 or less in the dominant eye [1]. However, we are not only considering the people in this category of visually impaired. Many people may only detect dark versus light objects, or may struggle to see with contact lenses or glasses on [1]. There is a large range of people who are considered visually impaired and encompassing everyone in all "trouble-seeing" categories takes up a large portion of the

North American population. In the United States alone, an estimated 20.6 million people fit into any of the above levels of vision loss, which is staggering [2]. The ability to create a device that will make travel more convenient and the device more portable in addition to giving the user a greater amount of feedback of their surroundings is something we strive for in this product. The device could truly help a great number of people, which is all we strive for with the VIA.

We hope to use our product to simplify the life of a person with vision loss. The VIA is a tool that can be used to "see" what is around the user.

2. System Specification

2.1 System Overview

The VIA is a handheld electronic device used to help the visually impaired maneuver through different obstacles in their day-to-day lives. This is done in three key steps: gathering data from the peripherals, processing the data, and sending audio feedback to the user when necessary.

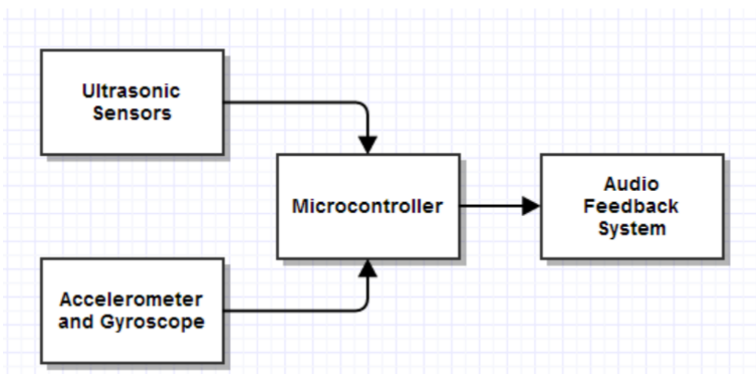

Figure 1 - Block diagram of the VIA's main components

The peripherals of which the data is gathered from are the four ultrasonic sensors, and a gyroscope. Between these five components, the microcontroller will have all of the data needed to accurately determine the location of obstacles in the direction that the VIA is being pointed.

The processing is done using an Arduino Uno[3], or an Adafruit Trinket Pro[4]. The microcontroller will be programmed to interpret and analyze incoming data to accurately distinguish between different obstacles the user may encounter. These obstacles will include, but are not limited to stairs, inclines, objects, and cliffs.

The audio feedback system is comprised of an external memory card, and the feedback system. There are three possible audio feedback systems: a mounted speaker, and a clip on speaker, and a headphone jack. The external memory card is needed to store sound clips that will be used to specify any oncoming obstacles. Also, if the clip-on speakers, or the headphone jack is in use, the mounted speaker will be turned off. A diagram of the VIA states is shown in *figure 2* below.

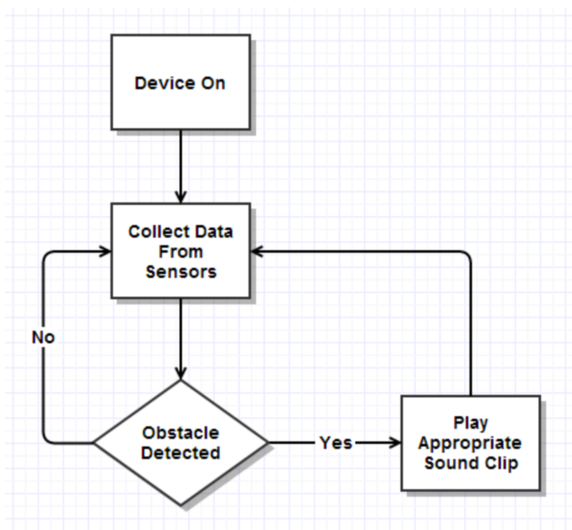

Figure 2 - State machine of the VIA

All of these components will fit into a compact, and ergonomic case. Making the device handheld allows for a smooth transition from the white cane while giving the user freedom to check for obstacles up to 6 meters in front of them. This encasement will also come with a sleek wrist strap to help avoid accidental drop damage, and misplacements.

3. General System Design

3.1 Parts Overview

3.1.1 L3GD20H 3-Axis Gyro Carrier with Voltage Regulator

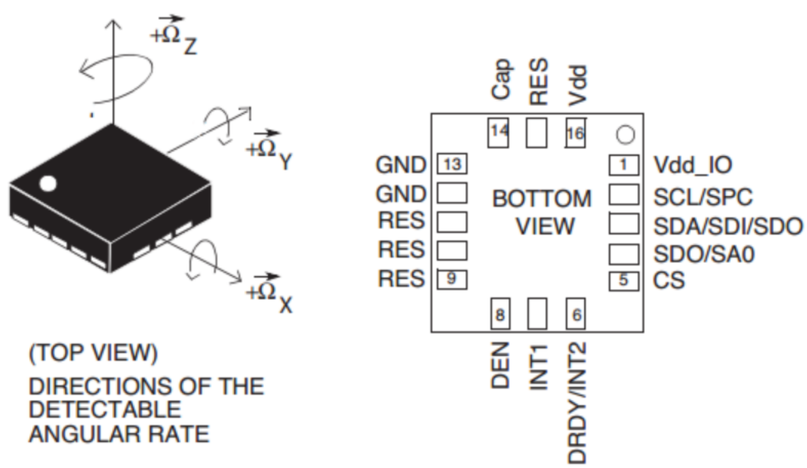

Figure 3 - L3GD20H Pin Connections

The L3GD20H 3-Axis Gyroscope, as shown in *Figure 3*, is a low-power three-axis angular rate sensor with an additional 8 bit temperature data output. It will be used to give accurate readings of the angles at which the user is holding the VIA. Under the assumption that the user will not be twisting the device in his or her hand, only one axis will need to be tracked. The reasoning for this assumption is that VIA encasement will be designed with hand grooves making it only comfortable to hold in a stable manner.

3.1.2 LV-MaxSonar® -EZ1™ High Performance Sonar Range Finder

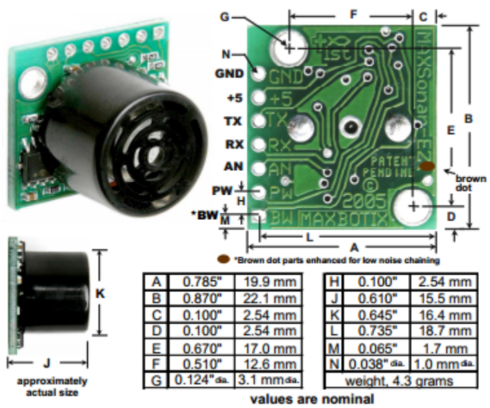

Figure 4 - LV-MaxSonar-EZ1 Dimensions

The LV-MaxSonar EZ1, shown in *Figure 4*, is an ultrasonic sensor that detects objects up to a distance of 6.45 meters away and with a 1-inch resolution. The output formats included are pulse width output, analog voltage output, and serial digital output. Using a scale factor of147 s/inch, we are able to get very accurate readings from the pulse width output method. For this to be done, only the GND, +5, and PW pins of sensor are needed. The main reason this sensor was chosen instead of other models is because of its size. Most other models had separate transmitters and receivers while the LV-MaxSonar combines the two. Also, our chosen ultrasonic sensor gives us both a max range, which is more than enough for our purposes, and sufficient resolution for a reasonable price.

3.1.3 The Microcontroller

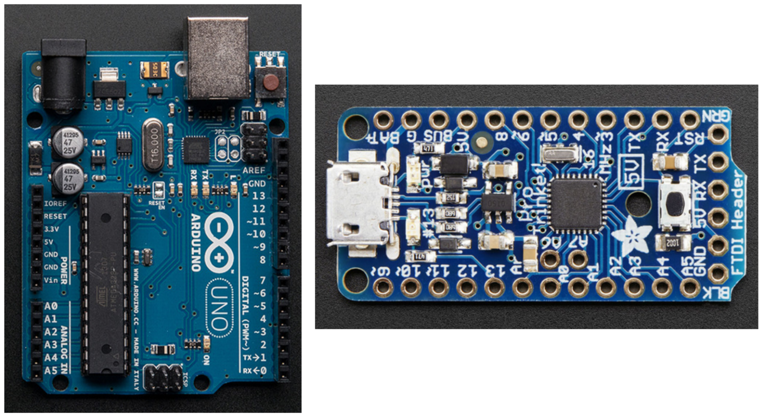

Figure 5 - Arduino Uno (left), and Adafruit Trinket Pro (right) microcontrollers

The Arduino Uno is a specialized microcontroller with 14 digital I/O pins, 6 analog input pins, and 32KB flash memory. Also, the fact that the Uno makes use of jumper wires instead of needing to solder wires on makes it a very flexible device that is ideal for prototyping projects. Because of this, all of the previously mentioned parts will be integrated into the Uno during the initial prototyping phase. As development shifts towards the final product, the Adafruit Pro Trinket will eventually replace the Uno. The Trinket is favorable to the Uno because of it's lower cost and smaller size. The shift in code from the Uno to the Trinket should theoretically be seamless because of how similar the two microcontrollers are.

3.2 Ultrasonic Sensor and Gyroscope Integration

The microcontroller will be programmed to take the inputs from the gyroscope and 4 ultrasonic sensors, and then determine if an obstacle has been encountered. A diagram illustrating the angles at which the ultrasonic sensors will be mounted is shown in *figure 6*.

As can be seen in *Figure 6*, θ_0, θ_1, θ_2, and θ_3 represent the angles that sensor 0, 1, 2, and 3 have been mounted on the VIA respectively. These angles are going to be fixed values. The variables d_0 to d_3 are the measured distance by the ultrasonic sensor 0 to 3 respectively. A diagram illustrating how the vertical and horizontal distances from points found by the ultrasonic sensors are measured can be seen below.

Sensible Solutions

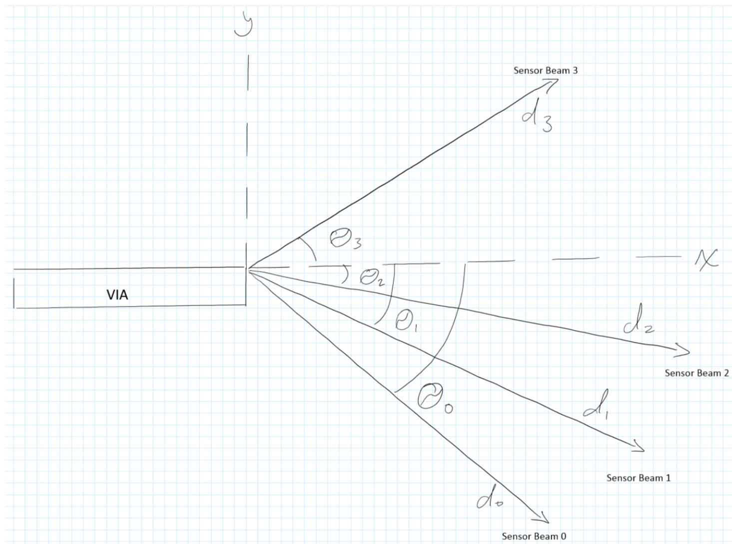

Figure 6 - Ultrasonic Sensor Mounting Angles

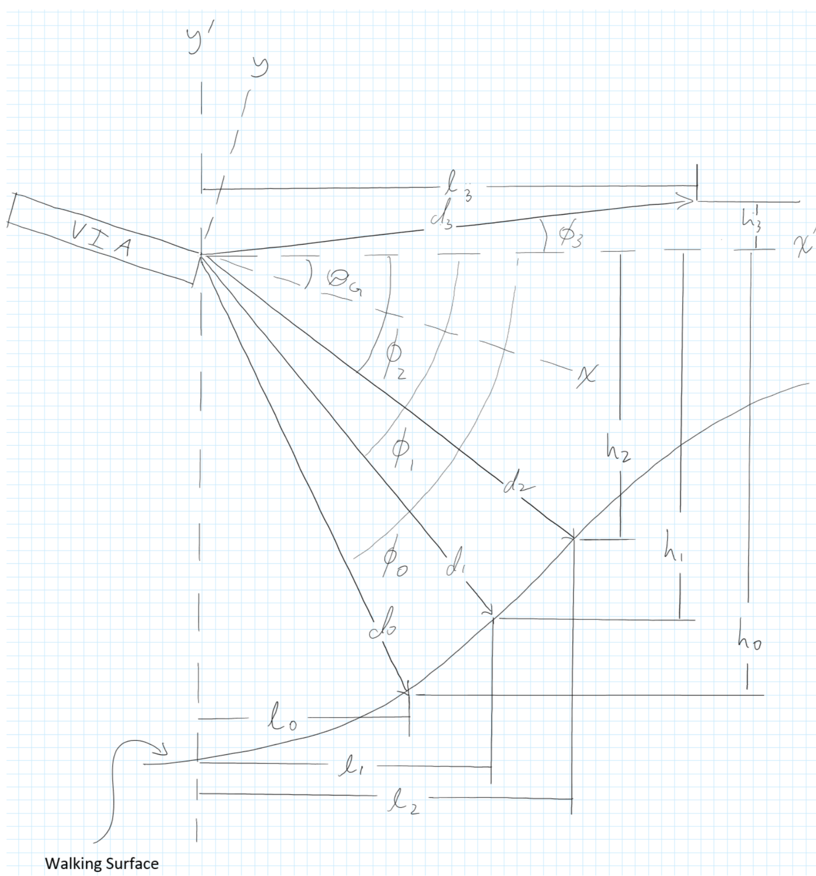

Figure 7 - Illustration of Object Detection

In *Figure 6*, the amount that the VIA has been angled down is $\theta_G = x'-x = y'-y$. This value can be found directly from the gyroscope. Using θ_G the known values of θ_0, θ_1, θ_2, and θ_3, we find the angles φ_0, φ_1, φ_2, and φ_3.

$$\varphi_0 = \theta_G + \theta_0$$
$$\varphi_1 = \theta_G + \theta_1$$
$$\varphi_2 = \theta_G + \theta_2$$
$$\varphi_3 = \theta_G - \theta_3$$

Ideally, $\varphi_3=0$ so that the top ultrasonic sensors beam is parallel with the *x'-axis*. Once these φ values are found, we can easily find the horizontal and vertical components of the distance measured by the ultrasonic sensors using trigonometric identities.

For instance, for ultrasonic sensor 0, we can find the horizontal component of the distance measured in the following way:

$$cos(\varphi_0) = l_0/d_0$$
$$l_0 = d_0 cos(\varphi_0)$$

Similarly, the vertical component can be found as follows:

$$sin(\varphi_0) = h_0/d_0$$
$$h_0 = d_0 sin(\varphi_0)$$

This method of finding component values can be used for each ultrasonic sensor distance measurement. A summary of this is shown in the following table.

	Angle (φ)	Horizontal Component (length)	Vertical Component (height)
Sensor 0	$\varphi_0 = \theta_G+\theta_0$	$l_0=d_0cos(\varphi_0)$	$h_0=d_0sin(\varphi_0)$
Sensor 1	$\varphi_1 = \theta_G+\theta_1$	$l_1=d_1cos(\varphi_1)$	$h_1=d_1sin(\varphi_1)$
Sensor 2	$\varphi_2 = \theta_G+\theta_2$	$l_2=d_2cos(\varphi_2)$	$h_2=d_2sin(\varphi_2)$
Sensor 3	$\varphi_3 = \theta_G-\theta_3$	$l_3=d_3cos(\varphi_3)$	$h_3=d_3sin(\varphi_3)$

Table 1 - Summary of Values from Figure 7

The values of h_0 to h_3 and l_0 to l_3 are used by the microcontroller to then determine if an obstacle has been encountered.
For instance, the simplest case of there being no obstacle and a level walking plane, $h_0=h_1=h_2$ and d_3 will be at its maximum (6.45m). On the other extreme, when encountering a wall, $l_0= l_1= l_2= l_3$. These two cases are shown in the *Figures 8* and *9*.

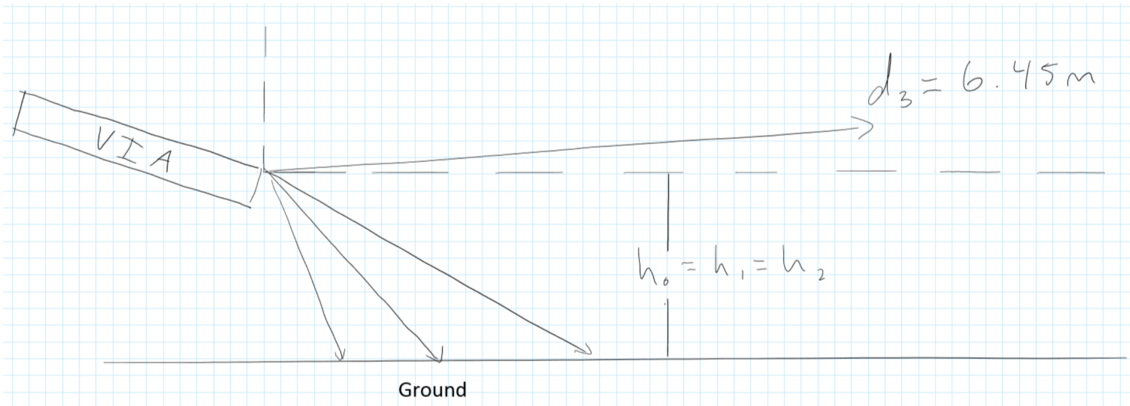

Figure 8 - Simplest Conditions for There to be no Obstacle Detected

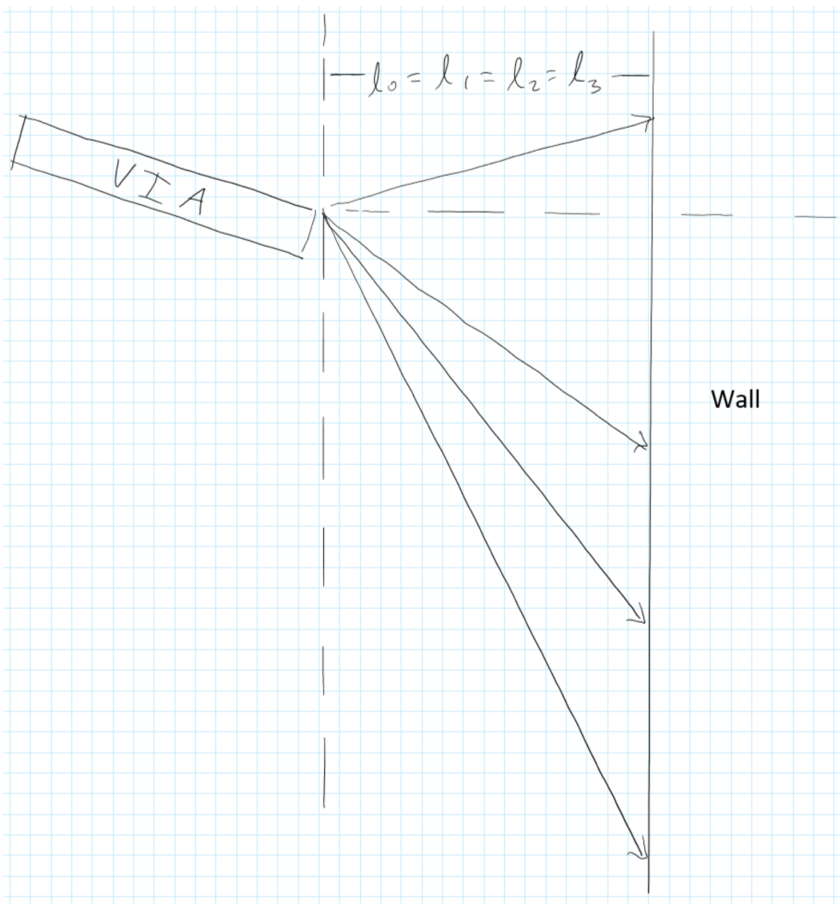

Figure 9 - Conditions met if VIA pointed towards a Wall

If the user was to be walking in a crowded area, he or she would need to angle the VIA further down to only sense obstacles in their direct vicinity. Depending on the angle θ_G that they hold the VIA, some sensors may need to be turned off. This is because at larger values of θ_G, some of the sensors could be pointed towards the users feet. To avoid getting these false positive readings, sensor 0 will be turned off when $\varphi_0 > 85$ degrees. Similarly, sensor

1 will also be turned off if $\varphi_1 > 85$ degrees. At these high θ_G angles, sensor 4 will behave similar to the way sensors 1 2 and 3 normally do, detecting obstacles on the ground instead of directly in front of the user.

4. Process Details

4.1 Functional Prototype

The functional prototype was developed around early to mid October. The prototype included the bare minimum in terms of peripherals and features - the three sensors at the head of the device, a buzzer notification system, and a distance-variance system. The prototype was created during the integration stage of the three head sensors.

The intended use of the prototype is to gain a better grasp of the ultrasonic sensors' functionality and interactions with multiple sensors, as well as to develop a building block for the obstacle-detecting algorithm. With a general layout of the device, we were be able to determine the optimal placement for current and future peripherals.

As we transition from the prototype to the working product, we will focus more on the unit development of other peripherals, namely the audio feedback system, and gyroscope. We also aim to implement the front-facing ultrasonic sensor during the later prototyping stages.

4.2 Prototype Refinement and Marketable Product

Our goal is to have a working product (prototype refinement) by mid November, and a marketable product by late November/early December. The development of the working product will begin with optimization of the peripheral placements to decrease interference, and move from using a breadboard to a printed circuit board. The most noticeable change in the product will be our transition from the Arduino Uno to the Adafruit Trinket Pro; this will give use a lot more leeway in the internal circuitry, while maintaining the same functionality of the microprocessor.

If the front-facing ultrasonic sensor has not been implemented at this point, that will be our first addition, bringing the total ultrasonic sensors in use to four. We intend to include most, if not all, of the peripherals which are required for the marketable product in the working product stage. The implementation of the audio feedback system, and gyroscope will be the main priorities.

During the development of the working product, our team's focus will be on peripheral implementation, and progressing the distance-variance system to an obstacle detection system. At this point, we hope that internal component interference will not be an issue so that we may focus on developing a better algorithm based on peripheral placement, instead of changing component placement to accommodate the algorithm.

Once said components have been integrated, we will begin the design of the device casing. The casing of the working product will aim to improve the aesthetics of the VIA; the ergonomic aspects will be improved during the development of the marketable product.

The marketable product will improve on the ergonomics of the working product, as well as fixing minor bugs in the obstacle-detection system.

4.3 Final Product

The production of the final product is highly dependent on our time constraints. It will be very similar to the working product; most changes will be minor aesthetic, and ergonomic changes.

There will also be additional bug fixes, and the gyroscope will be upgraded to an accelerometer. This upgrade will allow us to add a ‘cached’ warning system, instead of solely a real-time warning system. This cached warning system will allow the user to hold the VIA at a constant position, and warn the user of upcoming obstacle based on detection distance and the user’s walking speed.

5. Electronics Design

The main components are the VIA are the microprocessor, ultrasonic sensors, accelerometer, and audio feedback system (which in itself consists of multiple electronic components).

The microprocessor is the brain of the device, and will be interacting with all the device peripherals; during the prototyping stages an Arduino Uno will be used, while an Adafruit Trinket Pro will be used for the subsequent phases (the working and marketable product, as well as the finished product). The microcontroller’s interaction with each peripheral will be acknowledged in the individual peripheral’s subsection.

5.1 Ultrasonic Sensors

The VIA will use a total of four LV-MaxSonar-EZ1 Ultrasonic Range Finders; the sensors will occupy two digital pins, and one analog pin on the microprocessor. The use of two digital pins instead of analog pins is due to the lack of analog pins.

The two digital pins will be connected to the gates of PMOS transistors, with the source connected to V_{CC}, and the drain to the input of a 2-to-4 decoder, acting as the ‘select’ bits for the ultrasonic sensors. The output of this decoder will be connected to the RX pin, which acts as an ‘enable’ bit, of each ultrasonic sensor. The PW pins (to be explained later) of each sensor will be connected to a gate of a PMOS transistor, with the source connected to V_{CC}, and the drain connected to the inputs of a 4-input 1-bit multiplexer. The selection bits of said multiplexer will be the same selection bits as the 2-to-4 decoder. The multiplexer output is the input to the aforementioned analog pin.

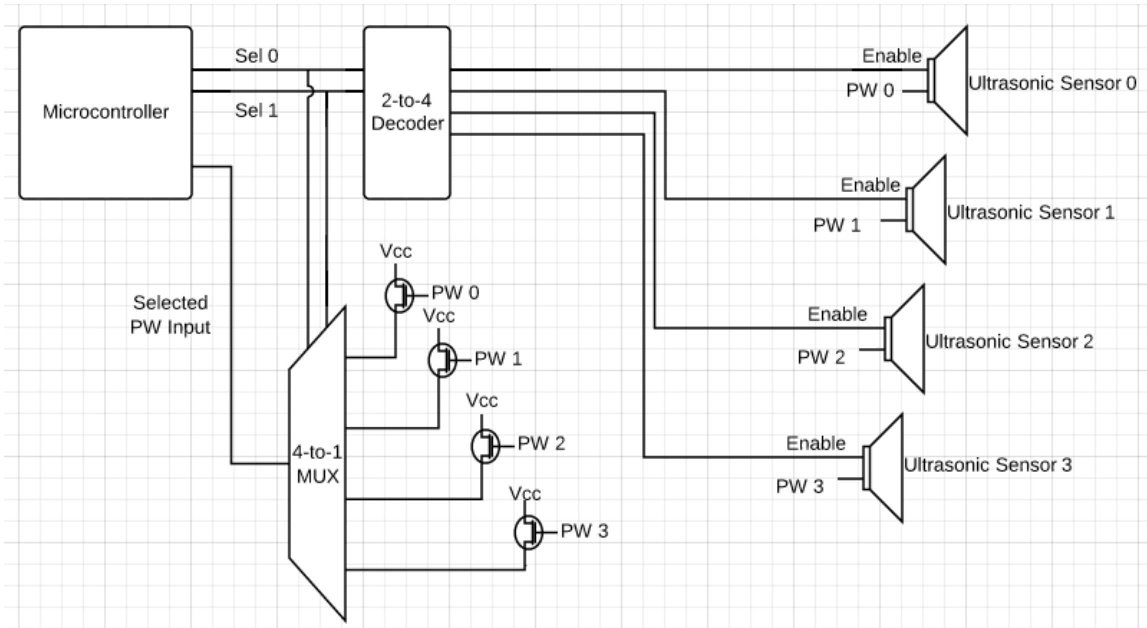

Figure 10 - Microcontroller and Ultrasonic Sensors Interaction

The use of the decoder and multiplexer is to decrease the pin usage, at the cost of a slightly more complex circuit and software. PMOS transistors are to circumvent the voltage drops of the decoder and multiplexer - as this proved very costly during testing. We also decided to use the PW pin (which is timing dependent, discussed later) to determine distances, since using the AN (analog) pin of the sensor is voltage dependent, which would be an issue, given the voltage drops of the MUX.

The PW, pulse width, pin of the ultrasonic sensors are used to measure the travel times of the wave. The time difference between two high PW pin pulses are proportional to the wave's travel time, the scale being 147 µs per inch travelled [5](easily transferred to metric).

Taking readings from the ultrasonic sensors require accurate timing to decrease the chances of creating interference from other sensors, as all ultrasonic sensors use the same frequency, 42 KHz.

5.2 Accelerometer and Gyroscope

Our intended progression would be to use a gyroscope in the working and marketable product, and an accelerometer for the final product. We agreed that this provides the best time investment-to-functionality trade-off, as the accelerometer requires software that is more complex.

Aforementioned, with the working and marketable products, we will be using a gyroscope instead of an accelerometer. The gyroscope will be using four digital pins, and will be utilizing the SPI, serial peripheral interface, Arduino library. This component interfaces well with the 3.3 V Trinket Pro, since its supply voltage ranges from 2.2 to 3.6 V.

The gyroscope measures the angle that the VIA is being held, thus allowing the microcontroller to determine the horizontal distance of objects detected, as well as preventing false-positives as mentioned in *Section 3.2*.

Time permitting, a 6-axis accelerometer will be integrated into the final product; this would not add to the complexity of the circuit as the microcontroller will have enough pins to accommodate the accelerometer. Both the gyroscope and accelerometer have the same input voltage requirement, 2.2 to 3.6 V[6][7], so we would not need to change the power source.

5.3 Audio Feedback System

The audio feedback system will be using an SD card to store the audio warning clips, since neither the Arduino nor the Trinket have the capability of storing audio files locally. The SD card module will be occupying three digital pins on the microcontroller, and will be utilizing the SPI library, similar to the gyroscope mentioned above. Transmission of the audio signal will occupy one analog pin and the RST pin.

For prototyping and unit testing we will be using an SD card shield; implementing a standalone SD card module will be seamless since the shield and module would be using the same pins and similar, if not the same, file access methods.

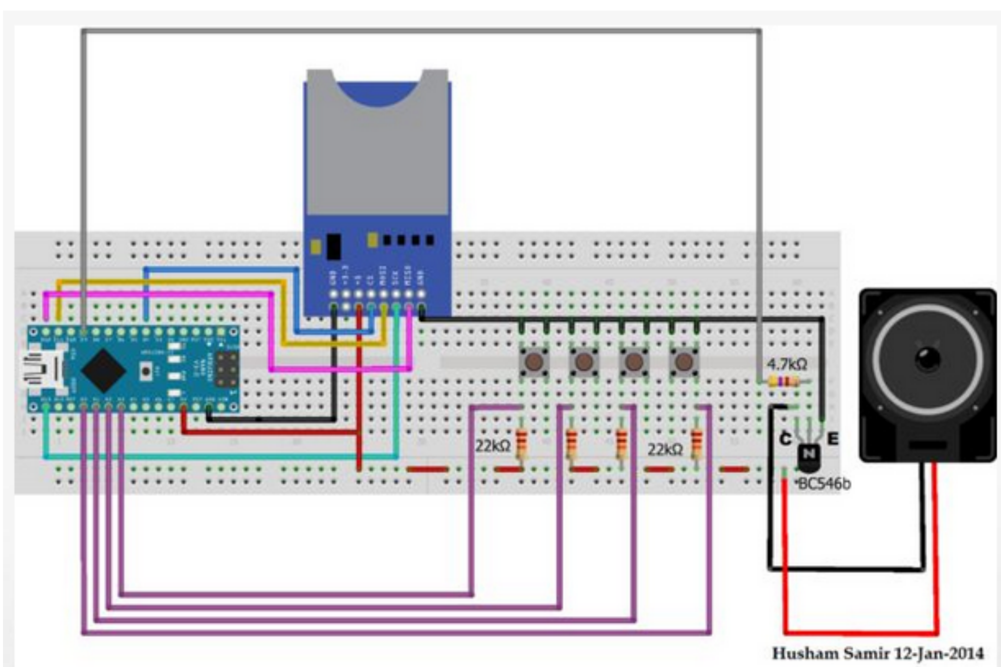

Figure 11 - Audio Feedback System Prototype[8]

As shown in the circuit on *Figure 11*, the speaker will not be interacting directly with the SD card module. The SD card module and speaker interaction will be moderated by the microcontroller, since it will handle all the file accesses from the SD card, and audio signal transmission to the speaker.

6. System Test Plan

It is very important that our system has as few errors as possible as our user's life is relying on this device. There are many dangers that can come from a person's surroundings, and awareness of those problems should be as precise as possible. We have discussed in the Functional Specifications the test procedures we hope to implement. However, we would like to use this Design Specification Document to outline the test plan in a more succinct manner.

6.1 Microcontroller Test Plan

The AdaFruit Trinket Pro will be constantly used throughout the implementation process. The testing of this component is extremely important for the function of our device.

- Blinking LEDs standard microcontroller test to ensure programs are loaded properly onto the board.
- Individual ultrasonic sensor implementation to identify any problems with each sensor.
- Multiple ultrasonic sensors implementation to identify errors with the sensors working together.
- SD card reader implementation to make sure SD cards can be read correctly.
- Speaker implementation to test audio output to the speaker
- Connection of specific components:
    - Multiple ultrasonic sensor readings to correct SD card output to check proper sensor reading triggers.
    - SD card to speaker to test the speaker outputs the correct sound clip.

Many of these tests will appear in other testing sections as well, as each of the components will be test individually on the Arduino Uno and then moved to become a part of the whole system on the Trinket. It is extremely important that each of these specific components work individually before they are integrated into the entire system. As well, once they become a part of the whole system, the device must be tested to determine any faults that appear with the new component implementation.

6.2 Software

The software implementation in this system cannot be overlooked and is one of the most important pieces of the product. Software will determine the integration of parts and how well each new component will trigger the next.

- Implement blink LED test program to determine if program loading works properly
- Check continuous data stream code for ultrasonic sensor. Must check that each individual sensor gives a continuous and correct sequence of distances.

- Implement program for three ultrasonic sensors working at once to determine if sensors interfere with one another through data stream. Check the code to correct any interference caused by the program as well.
- Fourth sensor will be the final sensor implementation and will be tested in software in the same way as the previous multi-sensor implementation.
- Audio Feedback System will be implemented in stages, as each component is necessary for the proper function of the system:
    - Stage 1: SD card reader finds a specific sound clip based on the output of the ultrasonic sensors.
    - Stage 2: SD card will play the specific sound clip through the speaker.
    - Repeat tests for each specific sound clip
- The gyroscope must determine the angle at which the user is holding the device and the accelerometer will determine the user's speed.
- The gyroscope will need extensive testing of proper software implementation as it is very possible to measure the angles incorrectly and give false readings in the data. The angle will be of great importance to the user as it will greatly affect the interpretation of the objects around them.
- The accelerometer software will also need extensive testing, as factoring in the user's speed is a task that will take large amounts of data and needs to determine if the data being measured is correct. A proper test would be to use a more reliable device (such as a running watch) to determine if the data found on our desktop matches a proper measurement tool.
- The LED will be simplistic in its testing with regards to the software. The software must check that the LED is turned on, which means the software component will be very small.

Software is another section of testing that will appear in multiple testing components as it connects many of the elements used together. Much of the software testing will happen in conjunction with the hardware testing, and it will be important to distinguish issues in the hardware component versus the software component.

6.3 Ultrasonic Sensors

The ultrasonic sensors will be the first peripheral component implemented on the microcontroller, and we are hoping for a smooth transition with each new sensor added. The main issue that could become a problem is interference with 4 sensors placed within a small region of the device.

- Ultrasonic sensor testing involves continuous distance readings that differentiate depending on the position of the sensor in relation to an object or wall.
    - Stage 1: test each sensor individually for reading errors.
    - Stage 2: Add a sensor, one at a time, to the configuration of sensors on the device. Test for each sensor and continuously check data readings.
    - Note: If an ultrasonic malfunctions, and a new sensor must be used, the new sensor must be tested for proper readings before it is implemented.

- It is important to test the multi-sensor implementation with each addition of a new peripheral that has a relation with the sensors. This will allow for few false errors in regards to the component currently being added to the system.

The sensors are vital to the system as their malfunctions could cause the user to feel the device is unreliable. As well, the greatest sense of surroundings will be the most desirable, which can only be achieved with properly functioning sensors.

6.4 Audio Feedback System

The functioning of the audio feedback system largely relies on the sensors giving correct outputs, and the software being implemented correctly. As discussed in the Software Test Plan section, the audio feedback will be done in multiple steps to ensure each portion of the audio loop functions properly.

- Stage 1: Implementation of the SD card reader and SD card.
  - Must use the board and software on the desktop to determine that the SD card reader is properly reading files from the SD card through the board.
  - Test the ultrasonic sensor interaction with the SD card reader, through specific files being chosen based on multi-sensor outputs.
- Stage 2: Implementation of the speaker system.
  - Use basic program to test if the speaker is giving an audio output.
  - Test connection to the SD card reader by checking if a single audio file will play from the SD card to the speaker.
- The final test is connecting the SD card reader, in conjunction with specific ultrasonic sensor readings, to the speaker with specific files playing through the audio device.

The stages of testing for the audio output should be implemented in numerical order as the speaker testing relies on the use of the SD card. Continuous testing should happen in each step of the audio feedback process as this is the final step in the creation of the functional prototype model. With all the components coming together, it is essential that multitudes of testing are accomplished.

6.5 Accelerometer and Gyroscope

The accelerometer and gyroscope are the first components that are regarded as not part of the proof-of-concept model. The hope is their implementation will not hurt the current model in place. The goal of these components is to allow the device to know its angle in relation to the ground and improve the distance calculation for the user in relation to an obstacle.

- Gyroscope:
  - Implementing and extensively reviewing the function of the calculations for the gyroscope within the code by orienting the device at different angles

  - If available, use a proper gyroscope to test the correct function of our gyroscope at different angles in relation to the ground. and whether the continuous data makes sense in relation to the ground.
- Accelerometer:
  - Continuous analysis of the a larger stream of data produced by the program written that determines the user's speed.
  - Comparison of the current user's speed to a running watch or other such proper acceleration measurement tool to determine if calculations of user's velocity are correct.
- Implementation of these parts to work with the ultrasonic sensors and testing of these features through the same tests as seen with the individual components above.
- Test the measurements of the gyroscope, as it should now factor in the device's positioning in relation to the ground. The test should consist of holding the product at many different angles and using the ultrasonic sensors to detect objects to determine if the sensors are still giving proper results.
- Test the measurements of the accelerometer, which now will use an assistive accelerometer to determine the user's current distance from an object. Determine through proper testing with more accurate devices if the values output for the distance have improved.

The accelerometer and gyroscope work with the ultrasonic sensors to give a more accurate depiction of the user and their environment. This means the testing procedure must be executed with as much continuous testing as was seen in the implementation of the audio feedback system. It will be affecting the functional prototype model from the base level, which means staying with testing procedures is vital.

6.6 LED External Warning System

This final component is not a key piece of the system, as it does not benefit the user directly. However, its implementation will allow the people surrounding the user to understand this person is visually impaired, as a warning system.

- Test the LEDs turn on when powered by the board.
- Add a small piece of code to the program to keep the light on at all times.
- Implement LEDs on the system board.

The LEDs should not affect the rest of the system, as the component has no relation to the function of the other components. However, it is still very important to test the model currently set-up on the board as a final measure to determine whether the LED implementation affected the rest of the system in any way.

This test plan has been thought through extensively, and we hope that it will allow our product to come out with few errors to make the transition to the users' hand as seamless as possible. It can be used throughout development to make a truly impressive product. To note, testing is extremely beneficial to the device as a whole and each stage of the testing cycle should be repeatedly tested in multiple ways to ensure a working model. This will allow for the greatest safety and reliability to the clients utilizing the product.

7. Conclusion

Sensible Solutions is well on its way to developing the VIA, or Visually Impaired Assistant; a device that will help the visually impaired navigate safely and confidently in their surrounding environment. This will be done using 4 ultrasonic sensors, a gyroscope, an audio feedback system, and an Adafruit Pro Trinket microcontroller. This design specification is written to allow the members of Sensible Solutions to create a product that accomplishes all of the goals outlined in the functional specification document. This includes how each of the features will be developed, and then tested. The test plan will allow us to find any unnoticed design flaws as soon as possible, giving us ample time to rectify the issues. Ultimately, the design specification allows us to streamline the development of the VIA into a truly marketable product that that will greatly improve the lives of anybody that is visually impaired.

Sensible Solutions

8. References

1. National Federation of the Blind. (2014, August 6). *Blindness Statistics* [Online]. Available: https://nfb.org/blindness-statistics [Accessed November 2nd, 2014]

2. American Foundation for the Blind. *Statical Snapshot from the American Foundation for the Blind* [Online]. Available: http://www.afb.org/info/blindness-statistics/2 [Accessed November 2nd, 2014]

3. Arduino. *Arduino Uno* [Online]. Available: http://arduino.cc/en/Main/arduinoBoardUno [Accessed November 3rd, 2014]

4. AdaFruit. (2014, August 20). *Introducing Pro Trinket* [Online]. Available: https://learn.adafruit.com/introducing-pro-trinket [Accessed November 3rd, 2014]

5. MaxBotix. *MB1010 LV-MaxSonar-EZ1 Datasheet* [Online]. Available: http://maxbotix.com/documents/MB1010_Datasheet.pdf [Accessed November 3rd, 2014]

6. STMicroelectroncs. *L3GD20H Datasheet* [Online]. Available: http://www.st.com/st-web-ui/static/active/en/resource/technical/document/datasheet/DM00060659.pdf [Accessed November 2nd, 2014]

7. InvenSense. MPU-6000 and MPU-6050 Product Specification Revision 3.1 [Online]. Available: https://cdn.sparkfun.com/datasheets/Components/General%20IC/PS-MPU-6000A.pdf [Accessed November 2nd, 2014]

8. Husham Samir. *Playing Wave Files using Arduino* [Online]. Available: http://www.instructables.com/id/Playing-Wave-file-using-arduino/ [Accessed November 3rd, 2014]