March 04, 2012

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

Re: ENSC 440 Design Specifications for Multifunction Intelligent Headphone System

Dear Dr. Rawicz,

Enclosed is a Design specification which describes the overall design of the Multifunction Intelligent Headphone System. The project is being designed and implemented with a system that will detect the voice in the surrounding environment in real time and process this information for the user. Some of the main features we are designing are word recognition, sound recognition, and voice recognition.

Our Design specification outlines the high-level design and testing of the intelligent system's hardware and software. This specification will be used by all team members in the design and testing stages. Targets are outlined in this specification for all members to ensure a minimum acceptable performance.

Our company, Sound Tec Inc., consists of five talented and innovative students from the School of Engineering Science at Simon Fraser University: Afrin Chowdhury, Frank Zhu, Leo Jiang, Simranjit Sidu and Xiao Peng He. We believe this team is capable of accomplishing the proposed design in a timely fashion.

If you have any questions about our design specification, please do not hesitate to contact us via phone at 778-855-4037 or email at jhj1@sfu.ca.

Sincerely,

Leo Jiang
Chief Executive Officer
Sound Tech Inc.

Enclosure: *Design Specification for Sound Tech Inc. Multifunction Intelligent Headphone System*

Executive Summary

The headphones in today's market provide great comfort and high-quality sound. These high-end headphones are completely noise proof. They not only eliminate noise, but also bring a lot of issues. These headphones can be very inconvenient to use when communicating with other people. They also bring danger to pedestrian users. Recent studies have shown that the number of injuries and deaths related to people using headphone on the street had increased [8]. The multifunction intelligent headphone system will provide users the comfort, sound quality and more safety.

The device will have three modes: word, sound, and voice recognition modes. In word recognition mode, the system will tune the volume up and down by detecting the keyword 'up' or 'down'. In sound recognition mode, the system will interrupt the headphone volume system and warn the user when danger sound is detected in the surrounding environment. In voice recognition mode, the system will inform the user if someone is calling user's name.

This document gives detailed specifications for the designing and implementation of the prototype model for Multifunction Intelligent Headphone System. Apart from the detailed hardware and software design, we have also documented the details of user interface flowchart, component breakdown, and testing parameters for headphone system. These design specifications will be designed and implemented in the prototype version of the end product.

Copyright © 2012, Sound Tec Inc.

Contents

Copyright © 2012, Sound Tec Inc.

List of Figures

List of tables

Copyright © 2012, Sound Tec Inc.

Glossary

CMOS:	Complementary Metal Oxide Semiconductor
CPU:	Central Data Processing Unit
DC:	Direct Current
ERX:	Serial Data Receive
ETX:	Serial Data Transmit
I/O:	input/output
LED:	Light Emitting Diode
MCU:	Microcontroller
MIHS:	Multifunction Intelligent Headphone System
NCH:	Noise Cancelling Headphone
PWM:	Pulse-Width Modulation
RX:	Receive
SD:	Speaker Dependent (SD)
SI:	Speaker Independent (SI)
SMVR:	SmartVR Voice Recognition
SPI:	Serial Peripheral Interface
TTL:	Transistor Transistor Logic
TX:	Transmit

Copyright © 2012, Sound Tec Inc.

1 Introduction

The Multifunction Intelligent Headphone System (MIHS) is a noise cancelling headphone that provides users great comfort, high sound quality, as well as safety. While providing a high quality of sound to the user, the MHIS will capture all the analog readings around the user, analyze them, and inform the user in the case of danger.

The headphone system will be equipped with a high-performance Arduino microcontroller board, which is used to decode the analog reading. The headphone has three modes: word, sound, and voice recognition modes. The headphone system will behave accordingly to the different modes. In word mode recognition mode, the MIHS tunes the volume according to the word command. In sound recognition mode, the MIHS will interrupt the headphone and signal the user when danger sound is detected. In voice mode, the MIHS will auto adjust the volume for the user upon recognizing certain voices.

The detailed design specifications for the MHIS are described in this document.

2 Scope

This document describes the design specifications for the Multifunction Intelligent Headphone System. All the hardware and software design details are covered. This document also specifies the design testing and user interface for the end product.

3 Intended Audience

The design specifications are intended for all members of the Sound Tech Inc. team to track the project progress and to be used as reference during the development of the device design. The team will use this document to ensure all the design specifications have been implemented.

Copyright © 2012, Sound Tec Inc.

4 System Specification

The Multifunction Intelligent Headphone System (MIHS) features three modes, each of the different modes will interrupt the headphone system and adjust the volume automatically when noise detected. User will have full control of the volume at any anytime, which is suitable for any mp3 or media player. The three modes are listed below:

- **Word Mode:** In word mode, the MIHS will detect any trained words and perform the action associated with the trained words.
- **Sound Mode:** In sound mode, the MIHS will detect any sound that matches the database and interrupt the headphone accordingly.
- **Voice Mode:** In voice mode, the MIHS will detect the voice of a specific and only perform action upon identifying the voice in the database.

4.1 System Overview

The MIHS consists of 4 main components, the noise cancelling headphone, the micro-controller, the voice recognition module, and the volume controller. The system's design specification and its interconnection of the four components will be discussed in the subsequent sections. A detailed explanation of the individual component and software component will be discussed in the hardware design and software design section.

4.2 High Level System Design

The MIHS is a feedback system and all the functions and tasks are controlled by the micro-controller. Noise being captured as input signal and processed to determine if certain action is required. The MIHS can be modeled at a high-level as shown in the figure below.

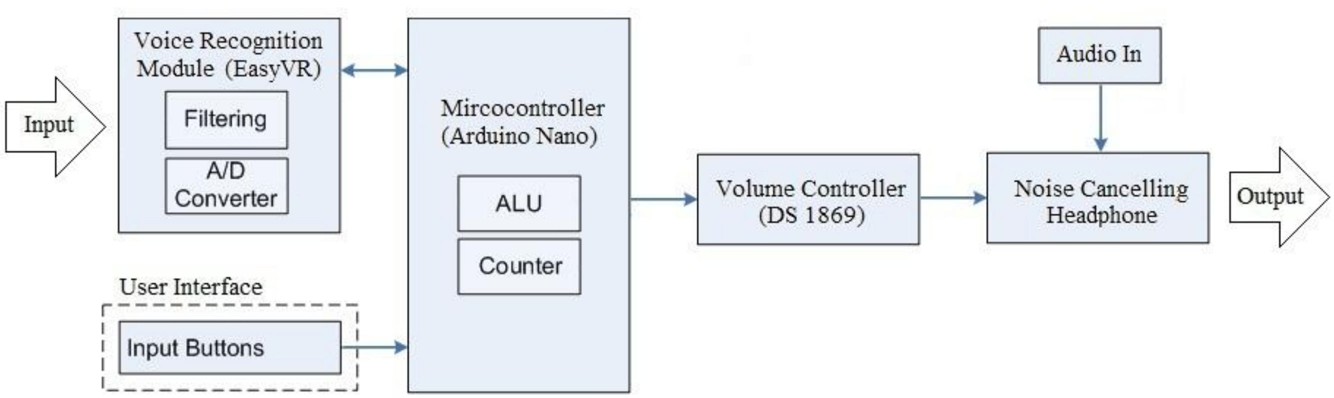

Figure 1: High Level Block Diagram of MIHS

Copyright © 2012, Sound Tec Inc.

From the block diagram, ambient sounds are being captured by the microphone of the voice recognition module, the EasyVR. The EasyVR will perform noise filtering and convert the signal to digital signal and sends it to the microcontroller.

The microcontroller has three main operations: checks the mode of operation, performs arithmetic operations, and sends pulse signals. The mode of operation is controlled by the switches, the three modes are: word, sound, and noise recognition. Once the microcontroller determined the mode of operation, it then performs arithmetic operation on the signal received from the EasyVR. It compares the data in the sound table that is stored in the EasyVR. If the signal matches those in the sound table it sends the number of pulse signals to the volume controller, the DS1869. The number of signal to send is controlled by the counter in the microcontroller, and it’s programmable.

The DS1869 is a digital potentiometer; the pulse signal from the microcontroller adjusts the resistance of the potentiometer. The varying resistance of the potentiometer results in higher or lower intensity of audio provided to the headphone.

The interaction of the different hardware is shown in figure 2. The figure represents the block diagram of MIHS that was introduced earlier.

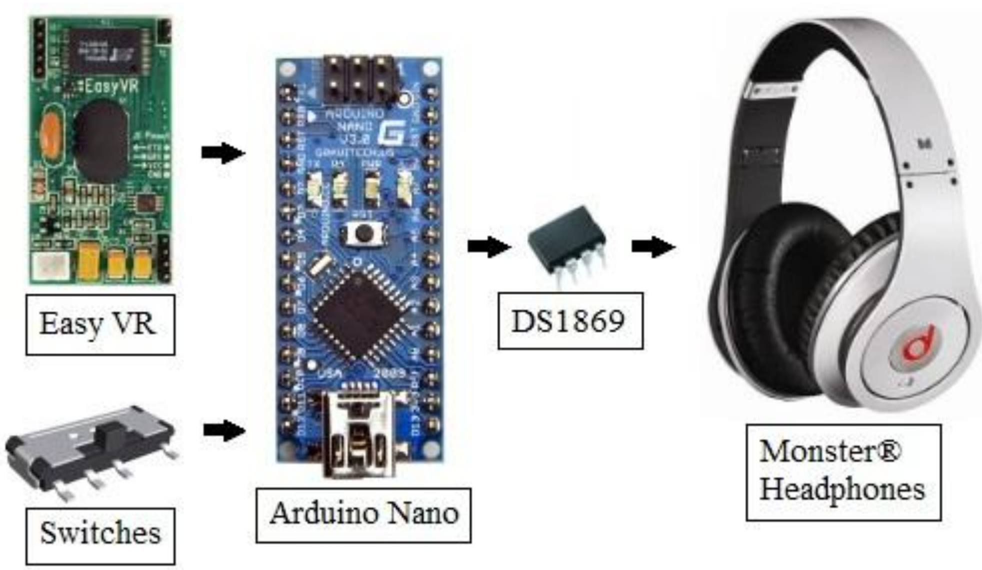

Figure 2: Component Interaction of MIHS

Copyright © 2012, Sound Tec Inc.

4.3 Component Placement

The placement of the hardware and user interactive features for MIHs is shown the figure below. The placement of the components will be for the proof-of-concept model and the final product, so the component placement design related to the functional requirement marked with II or III are taken in consideration.

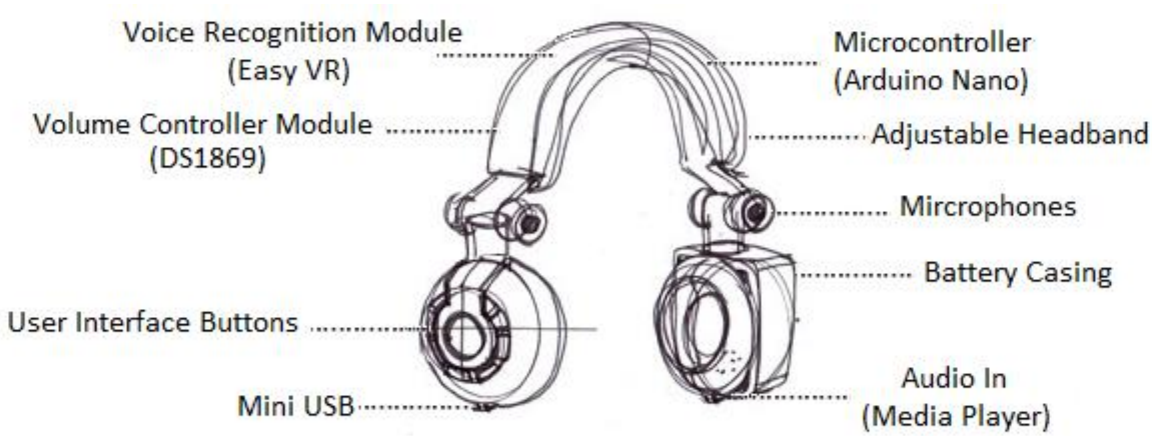

Figure 3: Component Placement for MIHS

From the functional specification, [R5-III] concerns the size of the MIHS and [R2-III] concerns that all circuits to fit inside the MIHS. There are two main circuit boards for MIHS, the microcontroller (Arduino Nano), and the voice recognition module (EasyVR). The dimension of the EasyVR is 1.77”x0.95” and the dimension of the Arduino Nano is 1.70”x0.73”, these two components can be fit inside the top portion of the headphone, inside the headband. The volume controller module (DS1869) is a small integrated circuit; it can fit alongside one of the main circuit boards.

The microphone is part of the EasyVR, it can be extended and be placed on top of the ear piece as shown in figure 3. Note placement of the microphones is directly above the ear when MIHS is being used, this placement essentially represents the ear and can pick up ambient noises.

The headphone and the circuits will require power to function; they are powered by the battery inside the battery casing located on one of the ear pieces. The other ear piece will be hosting user interface buttons, these buttons are for power, mode of operation, and reset.

At the bottom of the ear pieces, there is the mini USB input on one piece and an audio in on the other piece. The ‘mini USB’ is for communicating with the software to update the sound table of the EasyVR, and the ‘audio in’ is for audio playback.

Copyright © 2012, Sound Tec Inc.

4.4 System Circuit Design

The MIHS system circuit design is very simple, it is illustrated in figure 4 below. The detail of each of the pins will be discussed in the component hardware sections.

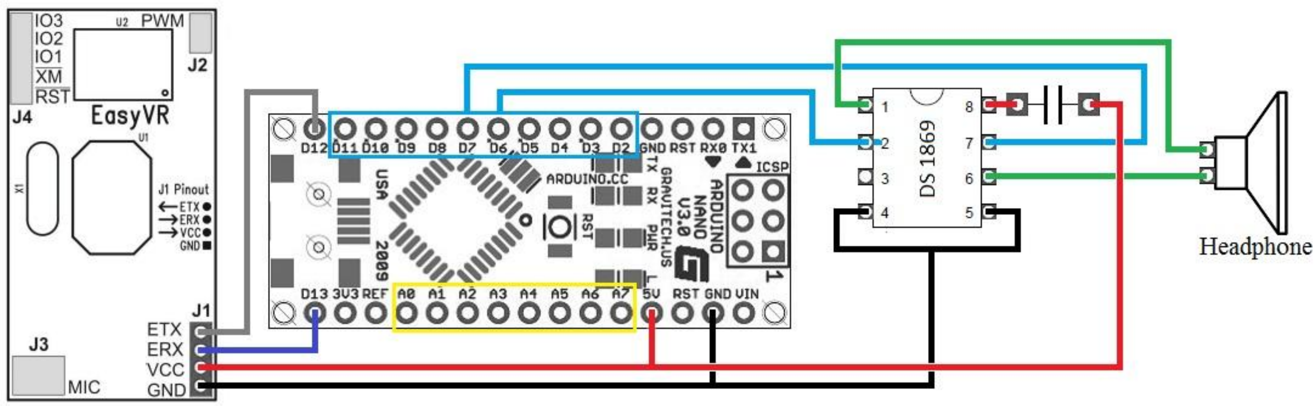

Figure 4: MIHS System Circuit Design

As shown from figure 4, the microcontroller is directly linked to the voice recognition and the volume control module. The headphone is directly linked to the volume controller and the audio in (not shown in the figure). There are two boxes highlighted. The highlighted blue box contains digital pins, the volume controller module can be controlled from any of the pins from the blue box, D2-D11. The highlighted yellow box contains analog pins, these pins will be connected to the user interface buttons, where mode of operation, reset, and power options are to be selected. The usages of the pins are determined by the programmer.

4.5 Electrical Requirement

The MIHS will have the battery as the main power source, this power source is rechargeable through the mini USB cable. All components will draw power from this battery source. The Arduino microcontrollers will regulate the voltage from the battery through its internal voltage regulator, this voltage is then being used by the voice recognition module, the volume controller, and the headphone.

The requirement for the batter power source of the MIHS must meet the following:

- Voltage: 5V (regulated)
- Maximum Current: 40mA
- Can be recharge via mini USB
- Regulated

Copyright © 2012, Sound Tec Inc.

4.6 Noise Consideration

Noise is a major factor for the Multifunction Intelligent Headphone System as large amount of noise reduces the quality of audio provided to the user. The design of MIHS must have minimal effects of the noise in the electrical system. The following steps will be taken to reduce the noise introduced to the system:

- Choosing high quality wires
- Routing wires efficiently to minimize the length
- Proper soldering of wires and pins

4.7 Safety Consideration

Electronic components in MIHS are sensitive to high voltage and currents. To prevent any of the electronic components from being damaged and prevent harm to the users the following precautions will be taken in place:

- Proper choice of battery is considered
- Heatsinks are used on electronic components
- Sensitive electronics are shielded

Copyright © 2012, Sound Tec Inc.

5 Hardware Design

5.1 Micro-controller design – Arduino Nano

The Micro-controller we are using in this project is Arduino Nano which is based on the ATmega328P processor from Atmel Corporation. Arduino Nano is the smallest, complete, and surface mount breadboard embedded version with integrated mini USB. Arduino Nano operates at 5V, has 14 digital I/O pins, 8 analog input pins, 16MHZ clock speed, a reset button, an integrated Mini-B USB Jack, and 32 KB programming and data memory space. The following figure shows the overview of Arduino Nano (ATmega328P).

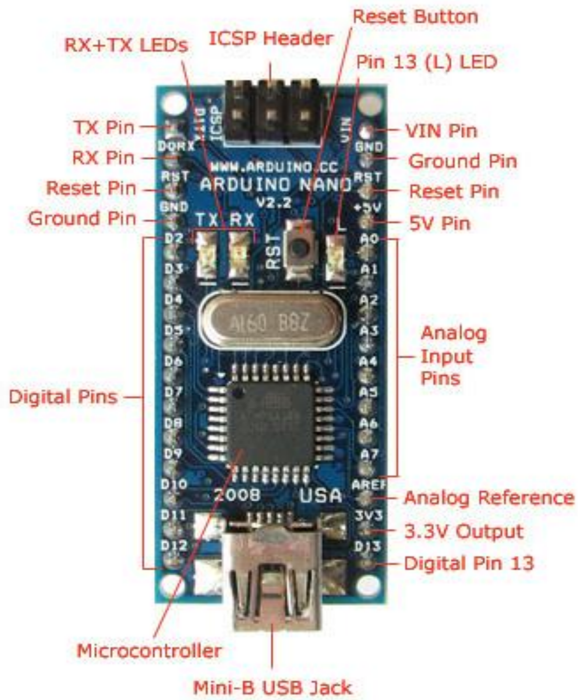

Figure 5: Top view of Arduino Nano [1]

By choosing different function, each of the 14 digital pins on Nano can be used as input or output. There are some pins have specialized functions such as:

- TX/RX pins for receiving/transmitting serial data
- PWM for providing 8 bit PWM output
- External Interrupts for triggering an interrupt
- SPI for supporting SPI communication
- Reset for resetting the micro-controller

Copyright © 2012, Sound Tec Inc.

The following two tables show the specifications of Arduino Nano and the ATmega328P.

Table 1: Specification of Arduino Nano [1]

Arduino Nano						
MCU	Operating Voltage	Input Voltage	Input Voltage(limits)	Digital I/O pins	Analog Input Pins	DC Current Per I/O pin
ATmega328P	5V	7-12V	6-20V	14	8	40mA

Table 2: Specification of ATmega328P MCU [1]

MCU	Flash Memory	SRAM	EEPROM	Clock Speed
ATmega328P	32KB	2KB	1KB	16MHz

The Arduino Nano can be powered by the USB connection and external power supply; higher voltage will be automatically selected for power source. Also, there are several ways that Arduino Nano can communicate with a computer, other Arduino board, or other micro-controllers. The Arduino Nano can be programmed with the Arduino software such as: Arduino Diecimila, Duemilanove, Nano w/ ATmega16, Arduino Duemilanove or Nano w/ ATmega32. All the details can be found in reference and tutorials from Arduino website.

Except the features of automatic reset during the program download, power OK blue LED, and ICSP header for direct program download, Arduino Nano provide an external manual reset switch.

Copyright © 2012, Sound Tec Inc.

The figure below shows the schematic for Arduino Nano:

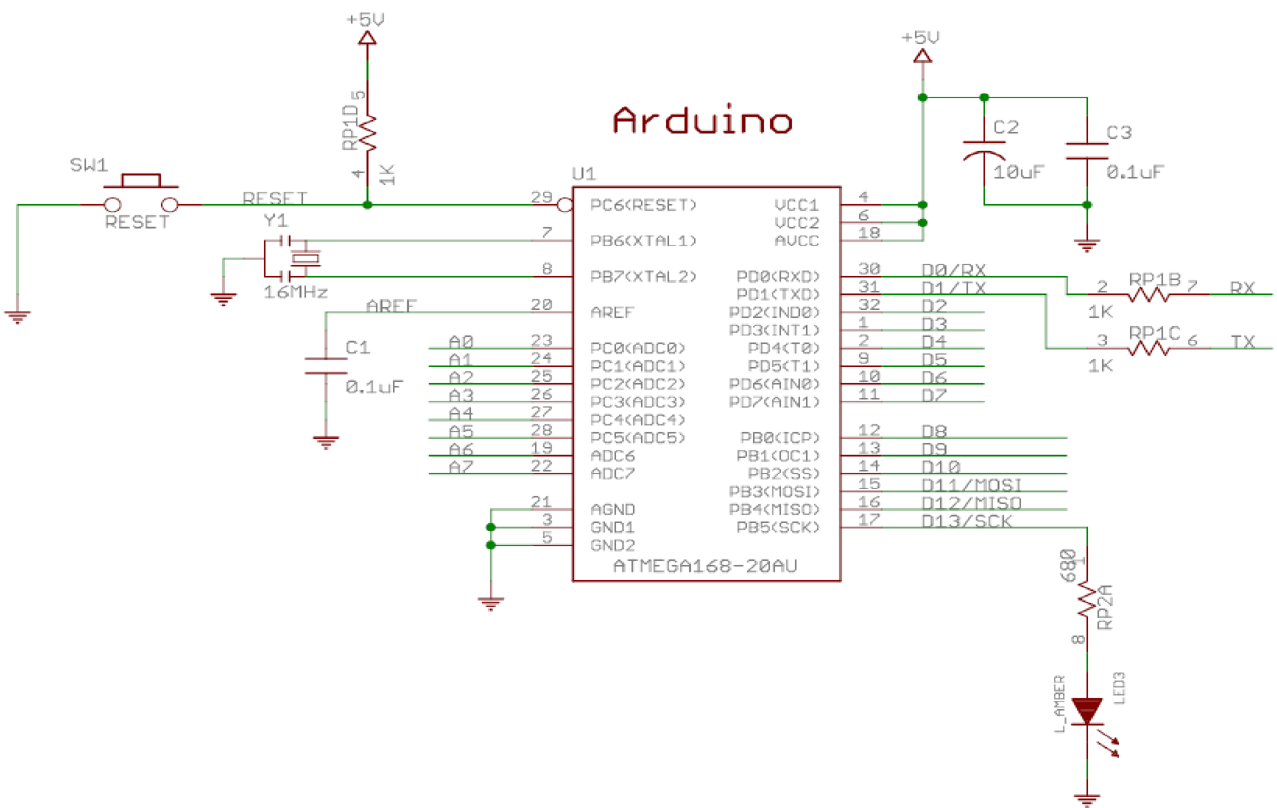

Figure 6: Schematic of Arduino Nano [2]

5.2 Voice Recognition Design – EasyVR

EasyVR as Voice/Speech Recognition module is the main component in the MIHS. It simplifies the design of speech recognition programs which integrates all the necessary components onto one board. Some of the key features of EasyVR are shown below:

- Excellent price (~$70).
- Appropriate Size (45 x 24mm).
- Controlled by built-in Speaker Independent (SI) commands in six different languages: English, Italian, German, French, Spanish, and Japanese [7].
- Supports 32 used-defined Speaker Dependent (SD) commands including voice passwords [7].
- It has simple Graphical User Interface program to perform voice commands and audio.
- It has the capability of sound playback of up to 9 minutes of recorded sounds and speech [7].

Copyright © 2012, Sound Tec Inc.

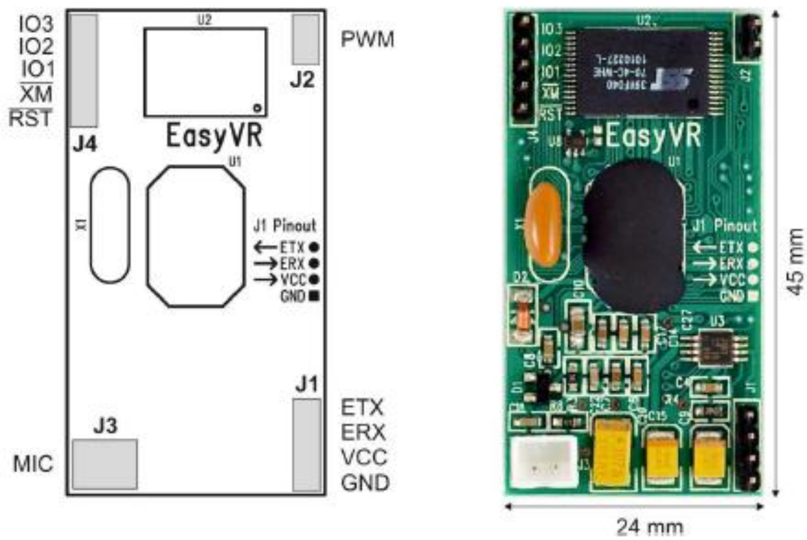

Figure 7: Physical diagram of EasyVR [7]

The EasyVR module operates at 3.3-5V, has four main connectors including I/O pins, microphone, reset button. The UART port includes bootloader which allows to update any flash firmware updates and to download new sound tables to the on-board memory. The approx. 2M bytes of flash memory will be used to store the source code and also the voice tags. Finally, the microphone will be used to receive audio input from the user, process it and send the command to the three I/O pins (IO1-IO3) which will be connected to the Arduino nano microcontroller. The Table below shows the connectors including pin assignments and pin description of EasyVR module.

Table 3: Pin assignment for Easy VR Module [7]

Connector	Pin Number	Pin Name	Pin Type	Pin Description
J1	**1**	GND	-	Ground
	2	VCC	I	Voltage DC input
	3	ERX	I	Serial Data Receive (TTL level)
	4	ETX	O	Serial Data Transmit (TTL level)
J2	**1-2**	PWM	O	Differential audio output (can directly drive 8Ω speaker)
J3	**1**	MIC_RET	-	Microphone reference ground
	2	MIC_IN	I	Microphone input signal
J4	**1**	/RST	I	Active low asynchronous reset (internal 100K pull-up)
	2	/XM	I	Boot select (internal 1K pull-down)
	3	IO1	I/O	General purpose I/O (3.0 VDC TTL level)
	4	IO2	I/O	General purpose I/O (3.0 VDC TTL level)
	5	IO3	I/O	General purpose I/O (3.0 VDC TTL level)

Copyright © 2012, Sound Tec Inc.

EasyVR module uses UART interface to communicate with the Arduino nano microcontroller. The UART interface is capable of 3.3-5V TTL/CMOS logic levels. Figure 8 shows the connection between EasyVR and Microcontroller.

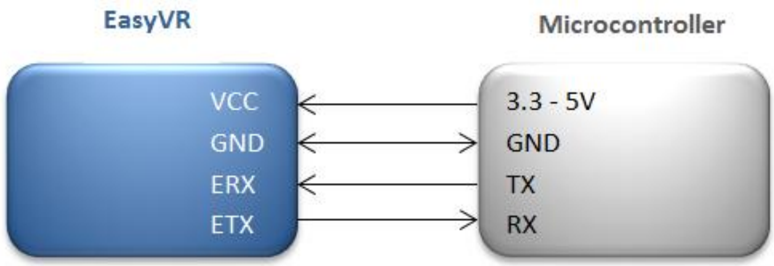

Figure 8: Connection between EasyVR and Microcontroller [7]

ASCII characters are used to communicate between EasyVR and microcontroller. Each command sent to the TX line and receives an answer on the RX line in the form of status byte. The delay of each byte sent out from EasyVR module to RX line should be at minimum. Initially the minimum delay set to 20ms and later on it selected in the range of 0-9ms, 10-90ms and 100ms-1s. For software based serial communication as EasyVR, very short delay might be needed before transmitting a character to the module. Serial communication is host driven, so each byte of the reply to a command has to be acknowledged by the microcontroller to receive any additional status data. If Microcontroller does not send all the required argument of command, the command will be ignored.

EasyVR is capable of both Speaker Dependent (SD) and Speaker Independent (SI) recognition. Speaker dependent is voice recognition technique where each word trained once and stored in one address. The speaker must train the system to recognize his/her voice prior to the recognition process. It has better accuracy than SI. On the other hand, the Speaker independent recognition needs to train each word four times and stored four different persons articulation of the same word to a different address with the same significant digit each time. It has lower accuracy compared to SD.

5.3 Volume Controller Design – DS1869 Dallastat™

We use DS 1869 Dallastat™ as volume controller of our project which is a digital rheostat or potentiometer. It can be controlled by either a switch contact closure input or a digital source input such as a CPU and it is offered in two standard IC packages which include an 8-pin 300-mil DIP and an 8-pin 208-mil SOIC. As an integrated circuit, the digital potentiometer replaces mechanical potentiometers with digital precision and it has no moving parts to wear out, saves space, reduces heat, and communicates with a processor for flexible and automatic adjustment [6].

Copyright © 2012, Sound Tec Inc.

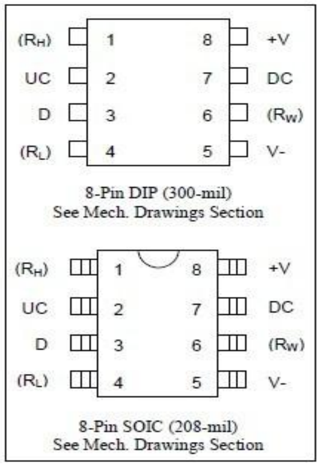

Figure 9: Diagram of DS 1869 [5]

The description of all the pins of DS 1869 is listed below:

R_H	High Terminal of Potentiometer
R_W	Wiper Terminal of Potentiometer
R_L	Low Terminal of Potentiometer
-V, +V	Voltage Supply Inputs
UC	Up Contact Input
D	Digital Input
DC	Down Contact Input

The volume control in the integrated circuit is controlled by a binary count up and count down. There are some key features such as [6]:

- Replaces mechanical variable resistors
- Electronic interface provided for digital as well as manual control
- Internal pull-ups with debounce for easy interfacing to mechanical push buttons
- Operates from 3V or 5V supplies
- Single 64- position potentiometer
- Wiper position stored in EEPROM
- Applications include volume, tone, contrast, brightness, and dimmer control
- 3V to 8V differential supply operational range
- Operating temperature: -40°C to +85°C
- 10 kΩ, 50 kΩ, and 100 kΩ versions

The following tables show the specifications of DS 1869 Dallastat™

Table 4: Specification of DS 1869 [5]

Digital Potentiometers						
Part Number	**Taper**	**POTs**	**Control Interface**	**Wiper Memory**	**Steps**	**$R_{END-TO-END}$ (kΩ)**
DS1869	Linear	1	Contact-Closure	Non-Volatile	64	10/50/100

Table 5: Specification of DS 1869 [5]

Digital Potentiometers					
Part Number	**Res. Tol. (%)**	**Temp. Coeff. (ppm/°C)**	**Wiper Resistance (Ω)**	**I_{cc} @5V (µA)**	**Smallest Available Pckg. (mm^2)**
Ds1869	20	750	400	10	44.1

5.4 Noise cancelling headphone

In order to experience the great detail of music and avoid the interfere by external noise, a noise cancelling headphone from Monster engineered Beasts by Dre Studio headphone will be applied in this project. The main features of this headphone include: advanced driver for precise audio clarity, high powered digital amplifier for avoiding distortion, and the active battery-powered noise cancelling technology for cutting external noise. Except the headphone cable and 1/8 inch adapter, the accessories also include two AAA batteries to activate the noise cancelling circuit. The following table shows the specifications of Monster Beats Headphone.

Table 6: specifications of Monster noise cancelling headphone [3]

	Impedance	Isolation	Connector	Headphone Type	Noise Cancelling	Microphone
Monster Headphone	1KHz 40 Ohms	-13dB~ -18dB	1/8 inch	Full Size	Yes	Yes

In the noise cancelling headphone, there is a small microphone that listens to the external noise inside each earpiece. Once the outside noise is detected by the microphone, the noise signal will be sent to the signal processing circuit to produce an inverted signal. So there are actually three signals which include the external signal, the inverted noise signal, and the music signal that are emitted from the headphone driver. The external noise signal and the inverted noise signal will cancel each other for their equal in amplitude but opposite in phase. So the only sound heard by ear will be the remaining music signal. The following graph shows how the noise cancelling circuit works.

Copyright © 2012, Sound Tec Inc.

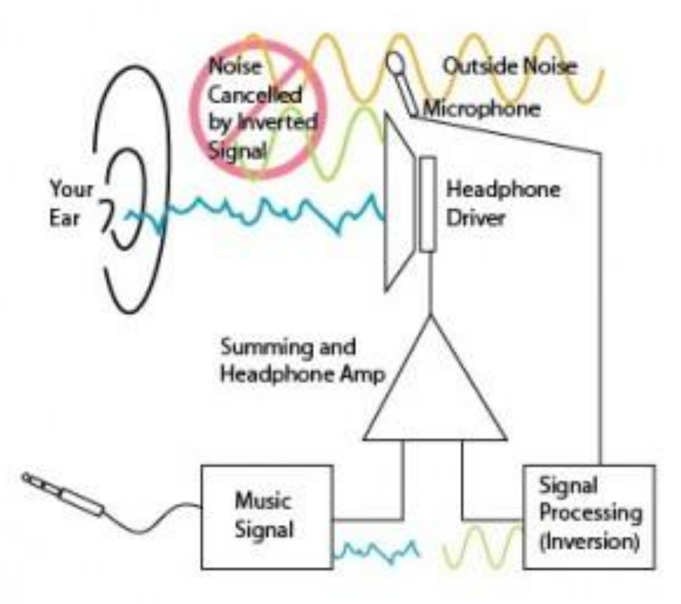

Figure 10: Noise cancelling diagram [4]

One important feature of headphones we need to consider is the Harmonic distortion rate. Theoretically, there is no harmonic peak for a perfect headphone that is impossible in reality. Blow the graph which is tested from the website, HeadRoom, where user can perform headphone measurements shows distortion rate of Monster noise cancelling headphone. The general understanding of harmonic distortion is that the distortion is less interfering when peaks decrease in amplitude while the frequency going up. The graph shows that the amplitude of distortion keeps decreasing as the frequency increasing.

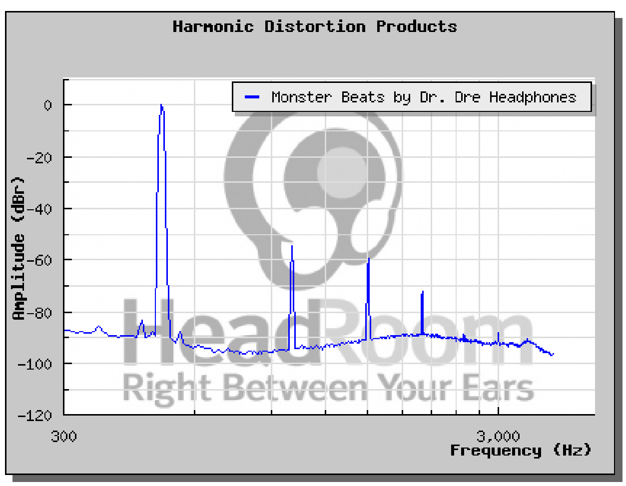

Figure 11: Harmonic distortion rate of Monster noise cancelling headphone [4]

The other test of Monster noise cancelling headphone is isolation measurement. The purpose of this measurement is to show the isolation performance that how well the headphone can isolate the external noise. The usual wisdom of isolation is that the ability of noise cancelling is better when the line declines more as the frequency goes up. So the next graph proves that Monster noise cancelling headphone attenuates the outside noise.

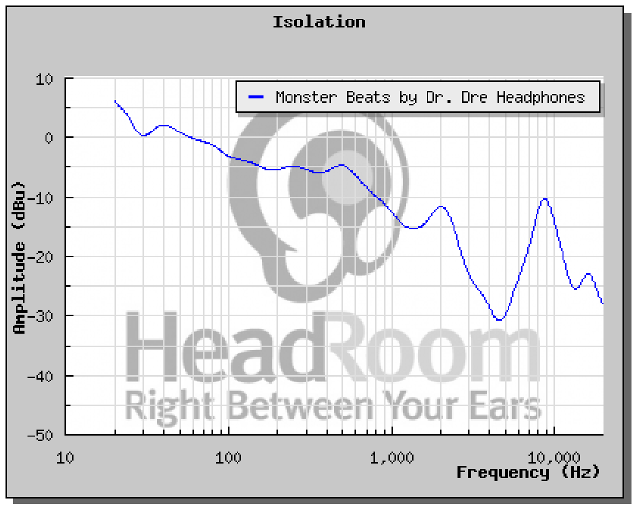

Figure 12: Isolation of Monster noise cancelling headphone [4]

5.5 User Interface Hardware

The user interface hardware consists mainly the buttons. As mentioned, the buttons are located on the side of the ear piece of the headphone. Each button is associated with a unique function; their functions are listed in the table below:

Table 7: User Interface Button Specifications

	Button Functions
Power	Turns the MIHS On/Off
Up	Turns the Volume Up
Down	Turns the Volume Down
Word	Put MIHS in Word Operation Mode
Sound	Put MIHS in Sound Operation Mode
Voice	Put MIHS in Voice Operation Mode
Reset	Reset after Word, Sound, or Voice Detected and Action Performed

Copyright © 2012, Sound Tec Inc.

6 Software Design

There are two software modules for MIHS, one is for the developer and the other is for the user. The software for the developer is the central software; it is also the control software of MIHS. The software for the user is for training words, sounds, or voices for MIHS.

6.1 Central Software System

The microcontroller is programmed using the Arduino programming language, a set of C/C++ functions, and the Arduino development environment. The program can be divided in three main parts: structure, values, and functions. The central system is programmed into the microcontroller; the central system contains a main process which calls functions for completion of tasks. The flow of the central system is shown in figure 13.

When MIHS is activated, the main process will run in an infinite loop to detect any input noise signals. When signal is detected, it will store it then call the mode function to check which mode of operation the system is set to. The three modes of operations are word, sound, and voice.

Once the mode of operation is determined, the main process will compare the stored signal to the database. The database contains the pre-trained word, sound, and voice needed for MIHS. If there is a match in the database, the system will process the data and sends output signal to the volume controller. The output signal is a set of pulse signals, the number of pulses is determined by the counter value in the program.

The pulse signals control the volume of the headphone, once the signals are sent to the headphone it will remain in idle mode until the user presses the reset button. The reset button will bring the whole process back to the beginning where it will wait and detect any input noise signals.

Copyright © 2012, Sound Tec Inc.

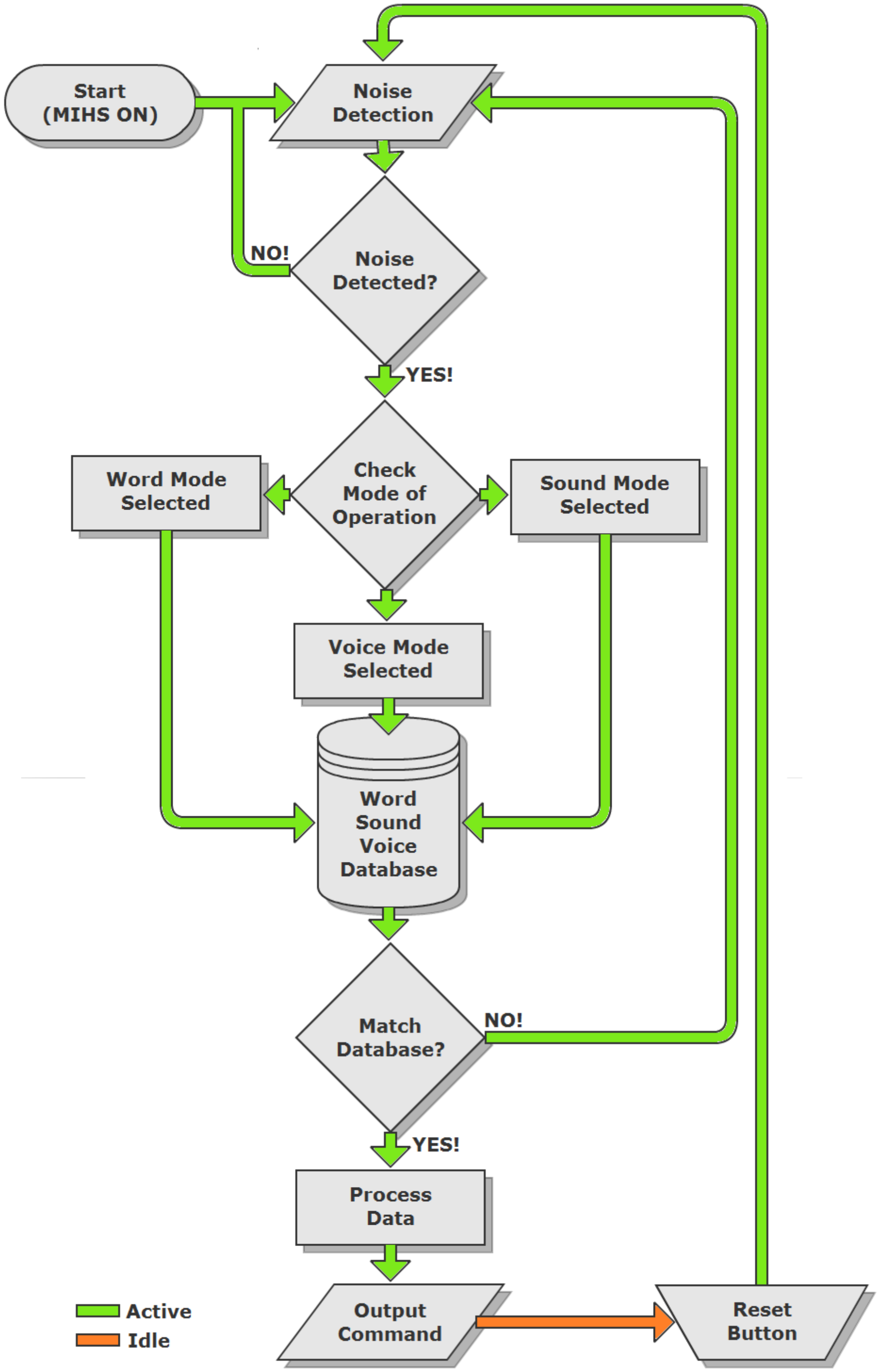

Figure 13: Central Software Flow Chart

Copyright © 2012, Sound Tec Inc.

6.2 User Interface Software

The Multifunction Intelligent Headphone System comes with software that allows users to customize the word, sound, and voice used. This is done through the EasyVR Commander. After the installation of the software, user can connect MIHS directly to their PC using the mini USB and start the EasyVR Commander.

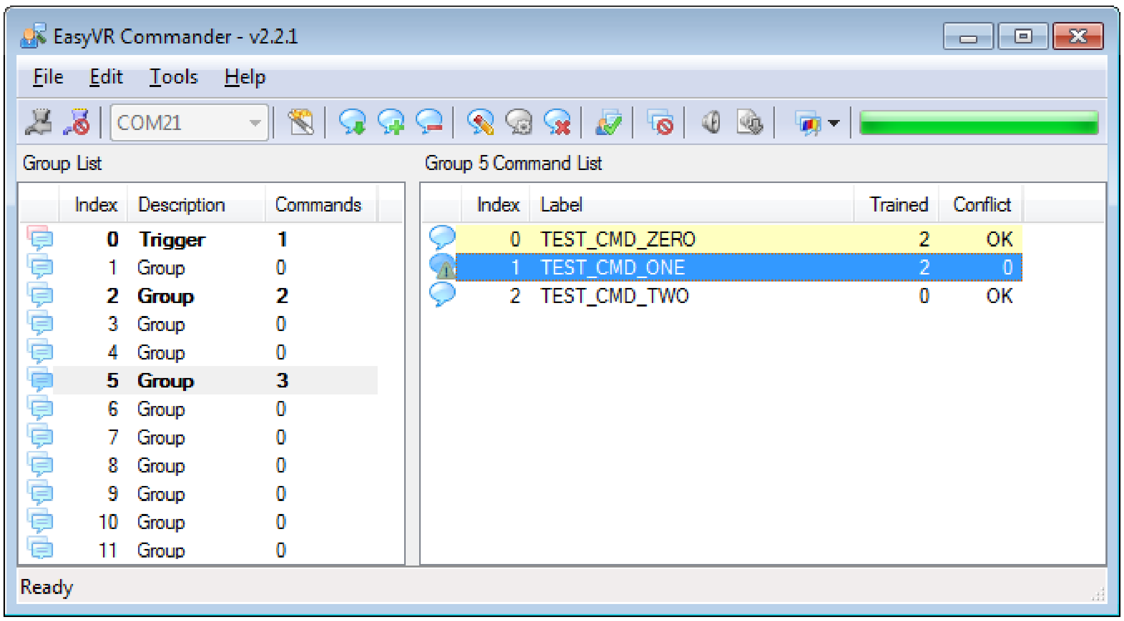

Figure 14: User Interface Software – EasyVR Commander

MIHS comes with pre-trained word and sound sets, simple word as up and down will trigger the MIHS to tune the volume up and down. It also comes with pre-trained sounds such as ambulance and police siren sounds. User can train special words such as 'one' and 'two' to tune the volume up and down, or add train more sounds. Using the EasyVR Commander, user can also train the special voice such as themselves or their parent's voice to trigger the MIHS.

7 System Test Plan

The MIHS will follow three in- depth test sections: Unit testing, Integrated testing, and System testing.

7.1 Unit testing

7.1.1 Microcontroller – Test Input/Output

Given a supply voltage of 5V to guarantee stable operation of the microcontroller, the voltage and current in each pin will be tested separately. By choosing different functions, each of the 14 digital pins can be used as input or output, so input/output will be tested naturally.

7.1.2 Volume Controller – Volume Up/Down

This device is a digital potentiometer which will be connected to microcontroller and headphone. We must test it whether can control volume to turn up/down.

7.2Integrated testing

7.2.1 Voice Module w/Microcontroller – Noise Detection/Action

The Voice Module is that EasyVR will be connected to microcontroller. Testing EasyVR is able to communicate with microcontroller, detecting noise from EasyVR and recording the data. Through analog signal of outside, the device will be tested sending the 'order' to microcontroller to deal with actions.

7.2.2 Volume Controller – Micro-control volume Up/Down

Testing DS 1869 Dallastat™ is able to communicate with microcontroller. Through the 'order' from microcontroller, the device will be tested achieving to control volume up/down.
System Testing

7.3 System testing

This section outlines test procedures for developers to ensure the proper operation of MIHS. The test plan consists of the following:

1. Test Mode Selection
    a. Power Mode
        i. Check if the "Power Mode" is properly connected to the necessary components in MIHS.
        ii. Check if the device is powered on and allowing user to listen music.
    b. Word Mode
        i. Check if the "Word Mode" button is properly connected to the necessary components in MIHS.

Copyright © 2012, Sound Tec Inc.

ii. Check if the “Word Mode” allows the user to detect any trained words by performing the required actions.
iii. Check if the “Word Mode” allows the user to detect any unknown word causing delay.

c. Sound Mode
   i. Check if the “Sound Mode” button is properly connected to the necessary components in MIHS.
   ii. Check if the “Sound Mode” allows the user to detect any sound by interrupting the headphone system?

d. Voice Mode
   i. Check if the “Voice Mode” button is properly connected to the necessary components in MIHS.
   ii. Check if the “Voice Mode” allows the user to detect specific voice and perform the action?

2. Test Word Control Volume
   a. Check if the device can detect any trained words accurately.
   b. Check if the device can automatically control the volume up/down.

3. Test Sound Control Response
   a. Check if the device can detect any surrounding sound.
   b. Check if the device can automatically control the volume down.

4. Test Voice Recognition Control
   a. Check if the device can detect any specific voice, such as user’s name.
   b. Check if the device can automatically control the volume down or mute.

5. Test Manual Volume Control
   a. Check by pressing the “UP” or “Down” button, if the volume is working properly.

6. Test Reset System/acknowledge Control
   a. Check by pressing the reset button, if the device is in default mode.

Copyright © 2012, Sound Tec Inc.

8 Conclusion

This document describes the overall design of our product, the Multifunction Intelligent Headphone System. It also discusses the design choices Sound Tech had made in order to meet the functional requirements. These design specifications will serve as reference for design and test team to realize the prototype of Multifunction Intelligent Headphone System. We believe to make a complete working prototype of the product ready by first week of April 2012.

Copyright © 2012, Sound Tec Inc.

9 References

[1] "Arduino," Arduino Nano Hardware Overview, 04 March 2012. [Online]. Available: http://arduino.cc/en/Main/ArduinoBoardNano

[2] "Arduino," Arduino Nano Manual, 04 March 2012. [Online]. Available: http://arduino.cc/en/uploads/Main/ArduinoNanoManual23.pdf

[3] "Beats™ by Dr. Dre™ Studio," Beats Noise Cancelling Headphone, 04 March 2012. [2012].[Online]. Available: http://www.headphone.com/headphones/noise-cancelling/monster-beats-by-dre-studio.php

[4] "Right between Your Ears," Noise cancelling circuit, 04 March 2012. [Online]. Available: http://www.headphone.com/rightbetweenyourears/?p=707

[5] "DS 1869," 3V Dallastat Electronic Digital Rheostat, 04 March 2012. [Online]. Available: http://www.maxim-ic.com/datasheet/index.mvp/id/2811/t/al

[6] "DALLAS SEMICONDUCTOR," DS 1869 3V Dallastat™ Electronic Digital Rheostat, 04 March 2012. [Online]. Available: http://datasheets.maxim-ic.com/en/ds/DS1869.pdf

[7] "EasyVR Speech Recognition Module," EasyVR user manual, 2012. [Online]. Available: http://www.robotshop.com/ProductInfo.aspx?pc=RB-Tig-01

[8] "More young males die in headphone-related collisions" CTV news, 2012. [Online]. Available: http://winnipeg.ctv.ca/servlet/an/local/CTVNews/20120217/more-young-males-die-in-headphone-collisions-120217/20120217/?hub=WinnipegHome

Copyright © 2012, Sound Tec Inc.