Capstone Design Abstracts

Fall 2015

Rover: The Pet-Sitting Robot

Team Members: Electrical Engineering: Alyssa Bezreh, Bryce Carter, Nick Newhart
Mechanical Engineering: Nick DeAngelis, Sienna Magee

Advisor: Professor Charles DiMarzio

Abstract

The Rover team has designed and built a pet-sitting robot to monitor and interact with dogs or cats left home alone by delivering treats and providing a live video stream to the owner. The robot (Rover) is controlled remotely through a desktop application which the user can install. Rover has the ability to maneuver indoors, monitor the pet, and dispense up to ten preloaded treats. According to our research, 36.5% of American households have a dog and 30.4% of households have a cat [1]. Rover provides companionship for these pets, delivers treats, and offers the owner a way to monitor them while away. While this idea has been proposed, there are currently no similar products available on the market.

The mechanical design of Rover includes three internal subassemblies and an external cosmetic shell. The first subassembly is the drivetrain which includes a chassis to provide structure as well as wheels and motors to drive the robot. The wheels are independently driven and bidirectional, enabling the robot both turn while driving and spin in place. The second subassembly is the treat dispenser which utilizes springs and a rotating shuttle to launch treats out of the robot. The dispenser is able to fire treats from two sides of the robot, allowing Rover to be fully reversable. The last subassembly is the camera, which can articulate 180º over the top of the robot depending on the direction of travel. The ability to move the camera up allows the user to see other parts of the room in addition to the path in front of the robot. There are four motors in the robot - two for the wheels, one for the camera, and one for the treat dispenser. The shell of the robot protects the internal components and electronics from both the environment and the pet. The robot is constructed from a combination of 3D printed parts, aluminum square stock, and an assortment of off the shelf and custom parts.

The electrical design of Rover includes hardware, firmware, and software components. For hardware, a custom circuit board was designed and fabricated. The board integrates the required voltage regulators with motor driver chips, sensor interfaces, and charging systems as well as connections with the microcontroller, a BeagleBone Black. Firmware was written in C++ with specific drivers for the camera, motors, and sensors. Linux device tree overlays were created to control the pin-muxing of the microcontroller. In addition, a self written networking library is used to accommodate communication between the desktop application and the Rover. Lastly, a graphical user interface written using the Qt Framework for C++ provides a simple method for the user to view the video stream and control the robot.

The Rover team believes the robot created is a unique solution which will solve a problem for a large number of pet owners.

sBox

A Carrier-Agnostic Delivery-Deposit System

Team Members: Luke Jaffe, Jefferson Akpona, Ran Tao, Xinhao Zhu, Yang Cui, and Eric Hu

Advisor: Professor John Kimani

Abstract

sBox was designed to meet the demands of online consumers who cannot always be present for package deliveries. sBox provides a secure system for receiving and storing packages while a consumer is remote. Several similar products are currently on market or in development, but are subject to flaws, including weak security infrastructure, lack of convenient digital UI, and unrealistic requirements for integration with existing postal systems. sBox is designed to solve these issues and define a realistic marketable product.

The sBox system consists of three discrete components: the secure delivery box (sBox), the central server, and the phone/web client for users. The physical sBox is designed to contain more than 95% of typical packages, based on Amazon package sizing and studies performed by competing companies. The box is meant to resemble a safe in security and appearance. The box door is locked using an electric strike system, powered by a custom charging circuit connected to a wall outlet. The electric strike lock is controlled by an in-house Android phone through an FTDI control board. The Android phone also serves as the box's interface to the remote digital system, and as physical UI for the package carrier and user.

The application running on this Android phone inside the box (box app), is a self-contained system which does not allow interaction with other aspects of the phone (kiosk mode). The box app is responsible for sending signals to the FTDI control board via USB through GPIO lines, and also to communicate with the central server through HTTPS POST requests.

One key aspect of our product is the barcode authentication mechanism, supported by the box app and the Android phone's front-facing camera. When a user places an online order, they record the tracking number, and input it to the sBox system through the phone app. When the package is delivered, the package carrier then scans the package barcode against the box app scanner (or manually inputs the tracking number if necessary), and the box app will unlock if this tracking number matches the one input by the user to the system earlier. The package carrier then opens the box, places the package inside, and the box closes and locks automatically. When the user returns home, they use the "unlock" feature of their phone app to unlock the box and remove the package. The phone app also allows the user to view order history, and receive notifications for delivery events and security hazards.

Testing of our system has confirmed the functionality of these features, in addition to several others. With secure installation, sBox provides an easy solution to the challenge of remote delivery for consumers, while also providing a simplified process for package carriers that may be unlikely to integrate with new overly-complex cyberphysical systems.

Jump-wire Free Breadboard

Team Members: Kaidi Du, Chisheng Liang, Kaitlyn Reilly, Gabriel Shaikh, Zhiyou Wang, Shenjie Wu,

Advisor: Professor Bahram Shafai

Abstract

Prototyping a circuit on a conventional solderless breadboard is a messy and error prone process. While most components are small and relatively sturdy, jump-wires can quickly clutter breadboards and are also very prone to pulling loose. They make problems difficult to spot and are generally a hassle to work with. And in order to reduce the need for wires, engineers often place components on a breadboard in a haphazard way that makes their circuits difficult to follow visually. The Jump-wire Free Breadboard (JWFB) aims to make prototyping circuits easier and more accurate by completely eliminating need for jump-wires.

Like an ordinary breadboard, the JWFB is organized as grid of tie points. All tie points in a row are electrically connected. To accommodate components in dual in-line (DIP) packages, the rows are laid out in two separate sections that are divided by a 0.2” trough. By inserting components into the proper tie points, a user can create a circuit.

Where the JWFB differs from other breadboards is in how it organizes connections between rows. On a regular breadboard, rows must be connected with jump-wires. On the JWFB, each row may be connected to any combination of eight available nets through an array of switches. By configuring this array of switches, an engineer can define and redefine the layout of a circuit. After the necessary components are placed on the JWFB, every aspect of implementing a circuit design can be achieved by flipping switches.

The JWFB’s power is derived from its organizational structure. Instead of forcing a user to organize a prototype around the predefined pinout of its largest component, the JWFB encourages the user to space components out. Components can be connected to nets instead of directly to each other. And for cases in which short leads are necessary, parts can still be positioned laterally like on a traditional breadboard.

While a user can configure the switch array by observation or by reading a netlist, engineers often base prototypes on SPICE files or netlists provided by schematic capture software. In order to make the JWFB more friendly to modern engineers, a custom software package was also developed. This software can interpret SPICE files and use them to generate a switch map. By following the switch map, a user can take a design that has already been implemented in software and bring it to the physical world with minimal effort.

Multisensory Brain Computer Interface for Binary Communication in the ICU

Team Members: Bertorelli, Laurel McCallister, James McLean, Sam Rendall, and Ginamarie Spiridigliozzi

Advisor: Professor Charles DiMarzio

Abstract

The CSL Capstone team designed a brain computer interface to facilitate communication between a caretaker and locked-in patients in the ICU. The primary goal of this project is to allow patients who are unable to communicate to undergo mental diagnoses for conditions like delirium, a mental health condition that significantly increases the risk of patient mortality. To prevent delirium, patients are screened three times a day in a process that involves asking a series of unambiguous yes or no questions designed to assess attention and clarity of thought. In severely disabled patients, these questions can only be answered with the help of a specialist through simple voluntary motions, such as hand squeezes or eye motions. Unfortunately, these methods are unreliable, and many patients lack the capacity for voluntary motion entirely. By allowing patients to respond to questions by simply focusing their attention, this project provides a means for even the most severely disabled patients to communicate with their doctors.

Patient responses are inferred by detecting Event Related Potentials (ERPs) and Stimulus Evoked Potentials (SEPs) in a patient's EEG signal, a measurement of the average electrical activity in the brain. ERPs and SEPs are modulated by attention, so by focusing on stimuli that correspond to either "yes" or "no", patients can control when ERPs and SEPs occur. A machine learning pipeline is then used to detect ERPs and SEPs and determine what the patient's response is most likely to be. This system is multi-modal, employing visual, auditory, and tactile stimuli to accommodate the diversity of injuries in an ICU environment.

The project consists of a stimulus system, a user interface, a data acquisition system, and a machine learning pipeline. The stimulus system uses a Beaglebone Black to issue stimulation command and a custom designed PCB containing an FPGA to drive the various modes of stimulation and assist with data acquisition. The user interface is written using *Qt*, and allows caretakers to easily control our hardware and run diagnostic tests. Data acquisition is carried out with a commercial ADC and a *MATLAB* library, which is used by machine learning algorithms, also run in *MATLAB,* to determine results. The system's components communicate through the *C++* networking library *OpenDDS* to allow each process to be run on a different host machine. In its entirety, this system can be used to complete the question and answer portion of a mental health assessment and record the outcome. The CSL capstone group is currently in the process of working with doctors to deploy this product in the ICU.

Nosferatu

Team Members: Connor Igo, Dan Cardin, Joe Gorse, Gunnar Boehm, Chris Holtsnider

Advisor: Professor John Kimani

Abstract

Nosferatu improves the lives of its users by adding convenience and removing annoyances from their daily routines. At times it can be inconvenient or even dangerous to stumble around in the dark looking for a light switch. Accidentally leaving the lights on when leaving the house leads to a more expensive electricity bill while leaving them off while on vacation can make the home seem abandoned and a target for burglars. Current wireless light bulbs are expensive and eventually need to be replaced. Nosferatu will replace a basic light switch with its own self-contained unit that will add improved functionality without needing to replace each individual bulb. This unit adds a motion sensor as well as remote control through a web interface. The motion sensor allows a user to turn on a light simply by walking into a room while the web interface allows them to turn their lights on or off from anywhere. The web interface allows the user to turn off their lights from bed before going to sleep or turn them on after waking up. While the user is out and wonders whether or not they remembered to turn off their lights they are able to check the status on the page and turn them off if needed. A user can also turn them on if they are not going to be home and want a light on to prevent their home from appearing empty. The web interface also allows a user to schedule times for their lights to turn on or off.

Each of the nodes is a NodeMCU board with a built in ESP Wireless Chip that can act as both an Access Point and Station, if the node has no active connection it will broadcast otherwise it will connect to the station. While broadcasting the node will just host a web page for a user to enter the SSID and Password of the station, which will then be used to connect and stop broadcasting. After the node connects to the station, the user is able to search for all nodes on their network and name them to better identify them later. From there the user is able to add rules to individual nodes. These rules can be set to change the status of a node automatically at a specified time. On the page the user is also able to disable motion control on an individual node. There is also a button on the node allowing manual control of the node’s status. Manual control of the node from the button or the web interface will automatically disable motion control for the node to prevent motion control from contradicting the user’s intentions.

In order to test Nosferatu, the team created multiple test circuits and connected one full circuit that would be found in a typical node to a desk lamp. A server was hosted locally on a Raspberry Pi. With this setup it was possible to easily test the motion controls as well as the button and control from the web interface. Testing of these features was met with success. Due to time constraints the Nosferatu team was unable to create a 3D printed case to house the circuit. Due to safety and time limitations the prototype was also not put to test in an actual home lighting scenario where the node is replacing a light switch.

Automatic Detection and Early Warning System for Oncoming Bicycles

Team Members: Robert Smookler, Samuel Sussman, Kongheng Wang, Shanali Weerasinghe, Mark Wilkening

Advisor: Professor Bahram Shafai

Abstract

The team has designed and built a portable system that alerts car drivers to bicycles and other vehicles in adjacent lanes. The system is intended to prevent accidents that occur when occupants of the car open their doors and strike cyclists who are currently in, or approaching, their "door zone." Serious injuries and deaths have been caused by drivers illegally opening their doors in the path of a passing cyclist where this is prohibited by law. The system does not replace the need for drivers to be aware of their surroundings, but rather acts as a security buffer, providing to the driver both visual and audio alerts indicating that it is dangerous to exit the vehicle.

The system implements two low-power 24GHz Doppler radar modules with asymmetrical narrow beams capable of detecting objects as far away as 100 meters. The two sensors are placed on the left and right side of the car near the rear windows, and fitted with an adhered tuned-frequency RF absorbent material that limits the sensor's field of view to its respective side of the vehicle. A moving object in range of the transceiver generates a low frequency output signal that is proportional to the detected object's speed. Each sensor is connected to an amplifier circuit and a bandpass filter that limits its bandwidth to a range that encompasses bicycles and cars - from 500Hz to 5000Hz. The frequency measurement is fed to the Arduino Uno microcontroller which handles signal processing in real time. A three-axis accelerometer assesses the current state of the vehicle. If the car is in motion, the driver is alerted by means of two LEDs placed on the dashboard. If the car is stopped, LEDs above the doors of each passenger will light up, and a speaker emits a tone indicating a cyclist or car is near. The system is powered off of one 12V lead acid battery, which is routed to a buck converter and then to the Arduino Uno. The microcontroller then provides 5V to all other components in the system. The battery is charged by the car battery through the 12V cigarette lighter interface and a boost regulator. Aside from the radar transceivers and their respective circuits, all elements of the system are housed within one module which is intended to be mounted on the dashboard of the car.

Testing and validation were performed in a controlled environment as well as on populated streets and highways. The results indicated that the transceivers are capable of accurately detecting oncoming cyclists and vehicles and alerting passengers of the car early enough to stop them from opening the door. However, the level of precision was unsatisfactory for two reasons: first, the "field of view" of the transceivers varied heavily depending on the orientation at which they were mounted in the car, and secondly because the frequency measurements became inconsistent when the car was traveling at the same speed as cars or bicycles behind it. It is the belief of the team that these issues could be rectified by performing more advanced signal processing and instructing the consumer on the best way to mount and install the system.

Visual Light Communication (VLC)

Team Members: Kyle Bradley, Ben Caine, James Croci, Ryan Nutile, Matt Schroer

Advisor: Professor Charles DiMarzio

Abstract

The VLC team has designed and built a comprehensive system for the wireless indoor broadcast of data using visual light. The proliferation of cheaper and more efficient LEDs has led to their integration into more and more facets of everyday life. In addition to LEDs increasing prevalence, visible light is not limited to the heavily congested wavelengths of the RF spectrum. Due to this, visible light is well positioned to emerge as an important data transmission medium. Our goal was to bring this technology out of the lab, where VLC researchers use high end Universal Software Radio Peripherals (USRPs), and create a functioning end-to-end system using commodity hardware and software.

The hardware system includes two separate components to transmit data wirelessly: the transmitter and the receiver. The transmitter receives a square wave signal at a designated frequency from one Beaglebone embedded computer, and uses this to modulate the red channel of a string of LEDs. The fast on-off switching of the LED creates a signal unperceivable to the human eye carrying our data. This data is then passed through an optical filter, converted into current by a photodiode, and into our receiver circuit. Our receiving circuit then filters, amplifies, and converts the received signal into a square wave to be read in by another Beaglebone embedded computer.

For software, data starts and ends in our Windows .NET GUI running on two computers, which can send and receive files, text, or webcam pictures over USB to our Beaglebones. On the Beaglebone's Linux partition we have a C++ application which is responsible for communication with the host computers, data encoding and decoding for forward error correction, packetization and depackization, management of physical memory, and control of the realtime units responsible for modulation and receiving of the signal. We then use the Beaglebone's two programmable realtime units (PRUs), which are each 200MHz microcontroller with full hardware access, to send and receive data using the GPIOs and share data with Linux via Direct Memory Access.

The resulting system is capable of transmitting data at speeds greater than 50kbps at about a meter distance, with potential for much higher distance and speed given any combination of electrical components better designed for high frequency operation, more robust forward error correction, higher optical power, or dedicated hardware for data encoding and decoding.

Coastal Automated Monitoring System (CAMS)

Team Members: Federico Beckhoff, Douglas Franklin, Benjamin Gowaski, Mitchell Kucia, Taylor Wilson

Advisor: Professor Bahram Shafai

Abstract

For people who live in coastal communities, the persistent threat of erosion is far too realistic. Threats which are exacerbated by particularly stormy seasons. Currently, the only methods available to collect shoreline erosion data are expensive and difficult to obtain. At most, data points are collected a few times a year. This leaves significant weather events, such as Hurricane Sandy which disintegrated New Jersey and New York shorelines in 2012, as disproportionally represented data points overshadowed by uniform weather patterns. We believe that coastal erosion is a momentous enough issue that better technologies meant to aid in data collection are urgently needed.

> *"Over the next 60 years, erosion may claim one of four houses within 500 feet of the shoreline."*
> *- Federal Emergency Management Agency (FEMA)*

To approach this issue, we first needed to establish which attributes are most directly related to coastal erosion. We quickly discovered this factor as the beach's profile or "slope" (i.e. on a more horizontal beach, the tide can run further up the shoreline). Soon after, we fabricated CAMS; a device that can be placed semi-permanently along shorelines of all shapes and sediment compositions, and measure all data deemed relevant by our research. Using the most current technologies such as LIDAR, tri-band transmissions (LTE), a Raspberry Pi for data aggregation and an Arduino-Mega board for motor and sensor control, we will be collecting a map of the beach's profile, as well as other meteorological information to determine which factors contain the strictest correlation. The LIDAR, a laser-based sensor that receives a distance to any obstruction it is pointed at is attached to a servo-motor, which moves on a semi-circular path allowing the LIDAR to trace the ground. Then, we can use the angle of the LIDAR's readings, with the distance reading, to paint an accurate graphical representation of the beach's shape using trigonometry. This process is to be run on a pre-determined interval, focused mainly around significant weather events, while the peripheral data (such as wind speed, barometric pressure, temperature, relative humidity) is also being collected. All the data will be stored in a database accessible through the internet for any scientific labs or independent researches that wish to study it.

The results of scientific analysis performed on this data could be utilized by municipalities, real estate agencies, plus construction and insurance companies who all have invested interest in understanding the patterns and likelihood of land loss and/or property damage due to erosion. Our biggest hope, however, is to provide peace of mind for at-risk communities by offering information which ensures more accurate preventative measures before any true danger arises.

Design of a Micro-EIT System for Characterization of Alveoli

Team Members: David English, Ibrahim Farah, Darren Lee, Jared Lowe, and Robert Mullinix
Advisor: Professor Charles Dimarzio

Abstract

In this project, an Electrical Impedance Tomography (EIT) system was implemented at a micro-scale in order to observe the impedance distribution in a small cross-section of an excised lung. EIT machines are currently used in hospitals on a macro-scale to monitor the functionality of a patient's lungs. The EIT machines can image the lungs because the air in the lungs have a much higher impedance relative to the lung tissue. This same idea applies to the micro-EIT system in that a higher impedance is observed in the alveoli inside of the lung. This system does not have alveolar resolution, but is used to observe a changing impedance distribution as the lungs and alveoli are inflating. The micro-EIT system has applications within a research environment to help researchers better understand the function of alveoli, especially in the characterization of lungs that are artificially inflated using a ventilator.

The standard set-up for an EIT system uses 16 electrodes arranged in a circular pattern around the chest cavity of the patient. An alternating current is then injected between two adjacent electrodes, and voltage values are taken simultaneously between the other 14 electrodes. This constitutes one image. Following this, the pair of electrodes with the current injection are rotated clockwise one electrode and the measurement process is repeated. The entire process results in 208 voltage measurements which are used to generate an impedance distribution by solving the severely posed inverse problem of correlating the voltage measurements to the currents in the targeted cross-section.

For implementation on a micro-scale, a similar process was used where the electrodes were placed millimeters apart in a circular pattern and then inserted into an excised lung. The digital and analog pins on an Arduino Mega were used to control the current injection and voltage measurements. The digital pins were connected to the select lines of two multiplexers and were used to control the current while the analog pins took the voltage measurements at the other electrodes. Stainless steel acupuncture needles were used as the electrodes and were placed in a fixture in a circle that penetrated the lung. This fixture was capable of being attached to an optical coherence tomography machine that was used to simultaneously take pictures of the first few millimeters of lung tissue as data was being collected. These images from the OCT helped to verify the data collected with the micro-EIT system. In MATLAB, an extension package of functions called EIDORS was used to interpret the data collected from the lung and create the impedance distribution.

Voltage measurements were taken at the same location on the lung while it was both fully inflated and deflated. Measurements were also taken in the same location at five different inflation intervals between the fully inflated and deflated lung. Using this system, it was possible to see the change in impedance as the lung was inflated. Again, it was impossible to attain alveolar resolution, but the fact that the change in impedance distribution could be seen has applications which would help optimize ventilating techniques and lung imaging.

BAT-PHONE: [B.A.T.M.A.N. Audio Transceiver Prototype Hybrid over Network Encryption]

Team Members: Benjamin Rosenberg, Derek Lavigne, Marija Vujasin, Sean Flaherty, Talya Sohnis

Advisor: Professor John Kimani

Abstract

The Bat-Phone, or B.A.T.M.A.N. Audio Transceiver Prototype Hybrid over Network Encryption, is a system consisting of a portable Wi-Fi audio transceiver, an intelligent mesh network, and a web application interface. The premise of Bat-Phone is an internet-sharing mesh capable of audio streaming over a network. An audio source is picked up via the microphone on the transmitter, and sent over Wi-Fi to a mesh receiver node. The audio can be streamed over the mesh network directly to a mesh node, or routed through an internet connected gateway node to a server. From the server, the audio stream can be forwarded to any internet facing device.

The main components of the transmitter consist of a microphone, an amplifier, a microprocessor running an embedded Linux OS, a Wi-Fi transceiver, and a Li-Ion battery with USB charging capabilities. The Bat-Phone transmitter has 2GB of NAND for storing Linux and our software, and 256MB of SDRAM for our scratchpad. The Wi-Fi transmits the packetized audio message over the mesh network to a receiver, such as a smartphone. The mesh nodes include different devices, such as routers, Mesh Potatoes, Raspberry Pis, and Bat-Phone transmitters that relay the messages through the mesh network to a respective client.

The Bat-Phone mesh network uses the B.A.T.M.A.N. Advanced routing protocol that operates on OSI Layer 2. A user can connect to the network as a mesh node or a non-mesh client. From a non-mesh client's point of view, the network looks like a conventional access point-based wireless LAN. Connecting to this access point will give a non-mesh client access to all of the mesh nodes. From a mesh node's perspective, the network consists of devices connected to each other through a wireless ad-hoc interface. If one mesh node has an internet connection, it can share internet access with all other mesh nodes. The Bat-Phone mesh network can operate without an internet connection through the use of "master nodes" that store information about the network, and can run a local copy of the web application and database.

The web app will be stored at the server under the domain batphone.co, and will be the user interface of our project. Its functionalities include registering user accounts, registering networks and devices, as well as controlling the infrastructure of the network. The web app includes maps showing each network's topology, with details about each device's functionality. Users of the web app will be able to record audio through devices, play live and recorded streams, and interact with other users on the network.

Use cases of the Bat-Phone range from simple audio streaming over a small mesh, to streaming over multiple nodes in a large network. The Bat-Phone could be used with one mesh node and a receiver in educational or office settings. The receiver could then record and share the audio stream from a meeting or lecture. In addition, many mesh nodes could be used in disaster situation to provide audio communication despite a lack of internet access or traditional communications infrastructure.

Discrete Intelligent Sphero System (DISS)

Team Members: Omar Al Mheiri, Mark Lee, Nelzir Louiseize, Rohini Rakhit, Sharan Sandhu, David Solomon

Advisor: Professor Bahram Shafai

Abstract

The DISS team has designed and implemented a system for universal security, which can be used in a variety of different terrains and situations with minimal set-up. This product will address the need for a module system that is discreet and small enough to function in multiple environments without the need to modify the current product or buy another product. This system has a wide variety of applications ranging from a home security system to a military hostage situation in which live video feed is needed to locate the hostages and to plan a course of action to resolve the situation.

DISS consists of three different parts, an Android application, a Sphero system, and a home base. To use DISS, the Sphero travels to the designated area by the path specified by the user through the Android application. In most cases, the Sphero will locate a base and travel to that area on its own, unless specified otherwise. The Sphero will then prompt the user to view a live video feed of that area and wait for further action. In a home security situation, the DISS Android application will prompt the user to dismiss the threat, contact police or other emergency services. The Raspberry Pi A+ was chosen based on its capability to easily stream video, its compact size, and its ability to provide a platform to easily connect and communicate with the Android application. Within the Sphero system is a separately designed battery and charging circuit setup, consisting of two 420mAh 3.7V Lithium Ion batteries in parallel, which powers the Raspberry Pi A+.

For testing and validation, two Sphero systems with two bases were designed and built, along with the development of an Android application. These systems were tested in a variety of locations, around the Northeastern University Campus as well as personal home locations to emulate a home security system. In battery testing, testing revealed that the Raspberry Pi A+ uses up to 700mA which we took into consideration for battery design. Early testing showed that the batteries would only sustain up to 5 minutes of video streaming which limits the usability for the applications originally proposed. It was also discovered that the additional charging coil, which is directly located beneath the Sphero’s original receiving coil, results in a longer charging time. Testing results indicated that our system could accurately locate and travel to a designated area and video stream to a user. However, when faced with unknown obstacles such as pets or furniture, our system was thrown off its course and had difficulty finding its home base.The DISS team believes that if a GPS system was implemented within our system, as well as a larger battery capacity to sustain functionality, DISS would be able to locate the home base efficiently and without failure. DISS is an easy and low-cost implementation of a surveillance and security system.

SoundSelect Array System (SSAS)

Team Members: Tim Deignan, Keenan Hye, Mark Long, Steve Muscari, Jack Tarricone

Advisor: Professor Charles DiMarzio

Abstract

The SoundSelect Array System team has designed and built a system capable of isolating sounds emanating from a specific point in a room and relaying that sound to a listener. The system can be "steered" towards the sound of interest to the listener, thereby reducing the relative gain of other audio sources. This system has a wide array of applications, including the fields of audio conferencing and television broadcasting. One application of particular interest is the use of the system as an aid to the hearing impaired. Hearing aid systems currently on the market are also designed to improve speech intelligibility, however they generally amplify all sounds in the speech band indiscriminately. This works in the near-field, since the closest, and therefore loudest, speaker is typically the desired sound. In the far-field, however, the lack of microphone directionality and a lessened signal to noise ratio combine to make it very difficult to understand a specific individual. The SSAS concept solves this problem by allowing the microphone array to be tuned to improve intelligibility of a specific target.

To use the SoundSelect Array System, an individual operates a control station connected to one or more microphone arrays positioned around a room. Analog audio signals from each microphone in the array are sent to specialized audio codec chips, which convert the audio streams to digital signals for manipulation. Using a technique known as Delay Sum Beamforming (DSB), the signals from the individual microphones are delayed relative to the angle of arrival of the target sound and summed back together, focusing a virtual detection pattern of the array in a specific direction. A ZedBoard containing a Zync-7000 SOC is used as a processor and the control station, with the actual DSB occurring within the ZedBoard's FPGA. This allows for processing at an extremely low latency, a requirement for using the system in real time. The control station also connects to a device for playback to the user, through either a hearing aid system or conventional headphones.

For testing and validation, a symmetrical array composed of five microphones was constructed using high sensitivity MEMS microphones. The audio signal from the array was processed separately in two frequency bands. Testing was conducted in a variety of locations around the Northeastern campus, including anechoic chambers, standard classrooms, and larger lecture halls. The results demonstrated the ability to target a specific desired sound source in a room. The degree of precision, and therefore the overall usefulness of the system, was a function of the number of microphone arrays, the number of microphones within those arrays, and the number of frequency bands which were individually processed from each array. While our results are already promising, the team feels the system could clearly be advanced by increasing each of those variables and using a larger FPGA to handle the system's increased processing requirements.