iSlide Avalanche Transceiver
Sr. Project Proposal

Jacob Sanders
Sean Jennings
University of Utah
# Table of Contents

- Project Description .................................................. 3
- Project Scope .......................................................... 4
- Implementation .......................................................... 5
  - Software .................................................................... 5
  - Receiver Circuit ....................................................... 5
  - Analog to Digital Conversion ...................................... 5
  - Digital Signal Processor ............................................. 6
  - iPhone Interface ...................................................... 6
  - Transmitter Circuit ................................................... 7
  - Potential Add-ons ..................................................... 7
- Bill of Materials .......................................................... 8
- Tasking and Project Milestones ...................................... 8
  - Transceiver Design .................................................. 9
  - iPhone Interface ..................................................... 9
  - Testing, Debugging, and Extending ......................... 10
  - Demo-Day ............................................................... 10
- Concerns .................................................................... 11
  - Digital Signal Processing ........................................... 11
  - Frequency Limitations .............................................. 11
  - Data Visualization ................................................... 11
  - Restricted Information on iPhone Hardware .............. 11
  - Radio Interference ................................................... 12
  - Cost ........................................................................ 12
- Conclusion .................................................................. 12
- Appendix A - Gantt Chart ............................................. 13
- References .................................................................. 14
Project Description

The intent of our project is to design an avalanche beacon accessory and application for the iPhone. Avalanche beacons are used by winter recreationalists to locate buried victims in the case of an avalanche. The device is worn by the user during outdoor recreation and runs in a low-power transmit mode, emitting a brief 457 kHz radio pulse about once per second. If an individual is buried in an avalanche, rescuers switch their beacons to receive mode and use them in conjunction with other search techniques to locate (and hopefully rescue) buried victims.

Originally, avalanche beacons simply transformed the input signal into an auditory tone whose intensity fluctuated proportionally with the strength of the signal, leaving all the signal processing to the ear and brain of the user. Most beacons on the market today implement a visual display and some form of signal processing to approximate the distance and direction to the victim. Our objectives are to understand how these devices work, design a compatible device, and build on the technology by interfacing with the iPhone.

Our interest in this project stems mainly from two sources. First, we share a common interest in embedded systems. This is essentially what motivated our microcontroller-based design over an analog design. Secondly, we’re interested in taking the software engineering principles we’ve learned to the wireless world through the development of a mobile application.

Researching this project has opened our eyes to some challenges and complexities that we had not anticipated. Given what little experience we have in signal processing, portions of the project seem daunting, but we are excited to take on the challenge and look forward to the opportunity to both demonstrate and expand our knowledge of the principles of electrical and computer engineering.

The essential hardware component of our project is a microprocessor-based transceiver circuit, and a 32-pin iPhone interface. The software component will include the programming of the microprocessor as well as the development of a user-friendly iPhone application. This proposal defines the scope of our project and outlines our anticipated design and schedule. We’ll also describe several risks and concerns as well as a vision for the future of the iSlide device.
Project Scope
The primary standard to which current avalanche beacon manufacturers adhere is EN 300 178, published by the European Telecommunications Standards Institute (ETSI) [1]. ETSI establishes mechanical and electrical standards for the devices as well as acceptable methods for testing their various components and functions. For the purposes of this project, we will not attempt total compliance with ETSI standards. We will, however, make our device compatible with others on the market by adhering to the following ETSI specifications:

- Operating frequency: 457 kHz
- Maximum frequency error: ±80 Hz
- Carrier keying (see Fig. 2):
  - On time: 70 ms minimum
  - Off time: 400 ms minimum
  - Period: 1000 ms ± 300 ms

![Carrier keying requirements of ETSI EN 300 716.](image)

ETSI also requires A1A Modulation (i.e. double-side band modulation for aural use without a modulating sub-carrier [2]. Unfortunately, this requirement places pretty serious restrictions on the functionality of the beacon since it doesn’t allow for data transmission via some other form of signal modulation. Since our project is not bound by ETSI standards as manufacturers are, we may choose to use a sub-carrier frequency in our design to enable digital data transfer between devices [3]. This, however, would be an added feature and is not essential to the scope of this project.
Implementation

Software
The software component of our project consists of a graphical user interface (GUI) and a device driver to interface our transceiver circuit with the iPhone. The driver will provide the necessary handshake before the transmission and reception of data can occur. Once the handshake has been established, the digitized signal information will be transferred to the iPhone where it will be interpreted and displayed on the GUI. The GUI also provides the option to transmit or search. If the user selects transmit mode, the screen will shut off to reduce power consumption and a signal will be sent to the transceiver unit, enabling signal generation. A double tap on the screen will re-enable the UI. This application will be written in a mixture of Objective-C, C, and C# code using the Cocoa framework for Apple iOS.

Because of the architecture of the iPhone OS, most iPhone applications are installed through iTunes and require a registration fee and a license of the developer. This is probably out of our budget and (as described in the risks section of this proposal) may be more work than its worth. We will probably need to disable the security of the phone (which requires some hacking) in order to create a non-developer application.

Receiver Circuit
The wavelength, $\lambda$, of a 457 kHz signal is approximately 656 meters. Obviously, a standard half-wave ($\lambda/2$) dipole antenna will be extremely large for our application. Fortunately, since the 457 kHz frequency lies in the medium frequency (MF) band right along side AM radios, we will be able to tune a standard ferrite rod AM radio antenna to the desired frequency using a tuning capacitor. The antenna we’re currently considering for the project is rated for frequencies ranging from 100 kHz to about 1700 kHz. The antenna is essentially an inductive coil and can be tuned using a variable capacitor to adjust the resonant frequency of the oscillator circuit as shown in Fig. 3a [7].

We will mimic the configuration of other devices currently on the market which use three antennae to determine the location of a buried beacon. This is accomplished by orienting the antennae in a mutually orthogonal pattern. The direction of the flux lines can be easily measured using basic trigonometry (Fig. 3b)[8]. We’re still researching the algorithms that will be used to compute the distance, but we anticipate that this calculation will be based on the signal strength at each of the three antennae.

Analog to Digital Conversion
We’re still researching how many digital bits of resolution will be required to adequately represent the 457 kHz signal at a reasonable distance from the source but in order to digitize the signal received from the antennae, each will need to be amplified to the appropriate operational range of the analog input ports of our microcontroller or other
ADCs we choose to implement. Aside from the antenna tuner, the amplifier should be the only analog circuitry that we need to design. We don't anticipate that this will be very difficult, but we'll probably need to break out the old Fundamentals of Electric Circuits textbook!

Since ETSI permits a frequency error of ±80 Hz [1], our amplifier should have a bandwidth, β, of 160 Hz. A principle in signal processing known as the Nyquist criteria requires that the sampling frequency of an ADC be at least 2β, or 320 Hz in our case. Sampling frequencies below the Nyquist frequency causes aliasing of the signal and necessitates an anti-aliasing filter. With a processor that will likely operate in the megahertz range, this shouldn't be an issue for us [5].

**Digital Signal Processor**
We have chosen to use a microcontroller unit (MCU) rather than analog circuitry to perform our signal processing simply because we are more interested in using a digital design. The MCU we're currently considering is the LeafLabs 32-bit microcontroller. It's overkill for our needs but will allow us the flexibility to add more advanced features to our device, time permitting.

The LeafLabs device uses the 72 MHz STM32F 103RB processor by STMicroelectronics and operates at 3.3 V (as does the iPhone). It has 16 analog input pins and two built in ADCs. The processor is rated for temperatures from -40°C to 85°C and supports low power modes that would be useful when the device is in transmit mode. This microcontroller is also fairly inexpensive. We plan to process the signals received from the antennae digitally on the microprocessor and then send simple packets of data to the iPhone (perhaps the angle and magnitude of a vector indicating the position of the transmitting signal relative to the receiver device).

**iPhone Interface**
The proprietary 30-pin iPhone connector has pins for USB and Firewire communication as well as two additional pins for serial data transmission and reception. We'll need to perform additional research before we determine specifically which pins to use, but we're currently planning on using the standard serial communication pins since we're not yet sure if the USB and Firewire ports can be used for any function other than synchronizing the iPhone device with a computer. There is also a pin on the iPhone connector that indicates to the iPhone what type of accessory device is connected to it. A serial communication accessory, for example, is identified by connecting this pin to a 500 kΩ load [9]. We will be using a PodBreakout board by Kinetica Systems [10] which will make it easier for us to solder wires to the iPhone connector (Fig. 5).
Upon plugging into the iPhone, a handshake will be made to enable the transmission and reception of data with the microcontroller. The phone will then send commands to the microcontroller and will receive packets of information from the controller with the processed signal data. The transmit mode will be activated by the user through the iSlide application which will send a command to the microcontroller to enable transmission. A 457 kHz frequency will then be generated (possibly modulated with additional data) and transmitted. If the user chooses to search, the iPhone will send a signal to the controller which will then activate the receiver circuit and relay information back to the phone.

Transmitter Circuit
We have several options available to us for frequency generation, but we're still looking into the method that will be most suitable for our design. There is a plethora of voltage controlled oscillators available, but most of them are designed for the mega/gigahertz range. We've reviewed data sheets for several chips in the kilohertz range but haven't yet found one suitable for our target frequency.

Crystal oscillators seem to be easier to come by in this range and 455 kHz is a common intermediate frequency for electronic devices, which may actually cause some problems for us as described in the risks section, but we are ignorant to the question of whether such a crystal could be tuned to work in this application.

The option that appears to be most promising at the moment is to generate the signal using our microcontroller, amplify it, and pass it to an antenna through a digital to analog converter (DAC).

Potential Add-ons
If time permits us to design a system that modulates the standard 457 kHz frequency with a sub-carrier for data transmission, we plan to identify the time of burial using the iPhone's accelerometer to detect decreased movement. If the built-in accelerometer is sensitive enough to discern heart beat and/or lung expansion, we will use it to determine the vital status of buried victims as well. Otherwise, we may choose to add a highly sensitive accelerometer or a heart-rate chest strap to the external hardware. This sort of data would then be attached to our carrier signal and transmitted for other iSlide devices to decode.
Bill of Materials

Several components of the project have already been purchased. Others will require additional research to determine the most suitable for our design, a partial bill of materials is listed below:

<table>
<thead>
<tr>
<th>Part</th>
<th>Source</th>
<th>Unit Cost</th>
<th>Qty</th>
<th>Total Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>LeafLabs Maple 32-bit MCU</td>
<td>RobotShop.com</td>
<td>$49.99</td>
<td>1</td>
<td>$49.99</td>
</tr>
<tr>
<td>Variable Capacitors, Op Amps, and Other Basic Components</td>
<td>ECE Stockroom</td>
<td>--</td>
<td>--</td>
<td>~$15</td>
</tr>
<tr>
<td>Apple iPhone</td>
<td><a href="http://www.verizonwireless.com/b2c/splash/iphone.jsp">http://www.verizonwireless.com/b2c/splash/iphone.jsp</a></td>
<td>$649.99</td>
<td>1</td>
<td>$649.99 (Purchased)</td>
</tr>
<tr>
<td>Avalanche Beacon (for testing)</td>
<td><a href="http://www.ebay.com">http://www.ebay.com</a></td>
<td>$120.00</td>
<td>1</td>
<td>$120.00 (Purchased)</td>
</tr>
</tbody>
</table>

Other components that may be required include the following:
- Voltage Controlled Oscillator
- Radio Crystal
- Digital to Analog Converter
- Radio Shielding for the iPhone (see the risks section)

Tasking and Project Milestones

We plan to finish as much of the core project over the summer as possible to allow for improved testing and further development. As we’ve begun to research the project, we’ve seen that extensive research is needed in two main areas, (1) iPhone-compatible hardware design and (2) digital signal processing. Though we would each like to be involved in every aspect of the project, we’ve decided that each of us will need to become the “expert” in one of these two areas. Jake will take the lead in iPhone research and Sean will focus on signal processing. We’ve identified the following milestones and tasks and have scheduled them as shown on the Gantt chart in appendix A.:
**Transceiver Design**

1. Analog Circuitry Design - Demonstrate a working receiver antenna and amplifier circuit using a test beacon to transmit a signal and an oscilloscope connected to the output of the amplifier.
   a. Tune an antenna coil using a variable capacitor.
   b. Design and build an amplifier that will receive a 457 kHz signal and amplify or attenuate the signal to the operating range of our ADC or MCU.

2. Multi-antenna Sampling – Using a logic analyzer, demonstrate that a signal generated by a test beacon is appropriately captured and digitized by three mutually orthogonal antennae.
   a. Connect analog circuitry to an ADC or the MCU. Verify using a logic analyzer (and a simple program on the MCU if needed) that an analog signal can be properly digitized.
   b. Build and test two additional analog antenna circuits.
   c. Write a simple program on the MCU to receive data from all three antennae.

3. Direction Algorithm – Demonstrate that the MCU is able to properly process data received from the three antennae and indicate the direction of the radio flux lines.
   a. Capture sample data from the receiver (and prepare additional simulated data if needed).
   b. Read test data and develop a mathematical algorithm in Matlab to compute the direction of the flux lines.
   c. Implement the algorithm on the MCU. Test using LED’s (or some similar method) to indicate direction.

4. Distance Algorithm – Demonstrate that the MCU is able to properly process test signals to determine the distance away from a transmitting beacon.
   b. Capture sample data from the receiver.
   c. Develop an algorithm in Matlab to calculate the distance.
   d. Implement the algorithm on the MCU using LEDs or an LCD display to indicate distance.

5. Signal Transmission – Demonstrate that our device is able to transmit a signal that is receivable by a test avalanche beacon.
   b. Design and test the transmitter hardware.
   c. Extend the MCU program to enable switching between transmit and receive modes.

**iPhone Interface**

1. Physical Connection – Demonstrate that the connector is recognized by the iPhone.
   a. Research the iPhone pin out and connectivity requirements.
   b. Solder appropriate wires and resistive loads to the iphone.
   c. Create a device driver on the iPhone if needed.
2. Create a Simple Program – Demonstrate a simple program on the iPhone making use of buttons and graphics.
   a. Jailbreak the iPhone.
   b. Install IDE and create a simple “hello world” application. Verify that it runs properly on the phone.
   c. Add buttons and graphics to the program to perform some trivial task.
3. Data Link – Demonstrate that we are able to transfer information (generated by a function generator and received by an oscilloscope) to and from the iPhone.
   a. Develop a simple application that receives a digital signal from a frequency generator over the Rx pin of the iPhone via the PodBreakout board.
   b. Extend the application to transmit simple data over the Tx pin. Verify proper transmission using a digital analyzer.
   c. Create a simple program on the MCU to transfer sample data to and from the iPhone.
   d. Establish a specific protocol for communication between the iPhone and MCU (i.e. how we will represent commands to transmit or receive or data such as the direction and distance of a received signal).
4. Graphical User Interface – Demonstrate a functional GUI version of the program.
   a. Design an aesthetically pleasing GUI.
   b. Finalize communication software and connect to GUI.

Testing, Debugging, and Extending
1. Quantitative Testing
   a. Test the devices compliance and deviation from ETSI standards.
   b. Test battery life.
   c. Perform spectrum analysis.
   d. Test signal strength and range.
2. Qualitative Testing
   a. Test device for usability and accuracy in a variety of indoor and outdoor environments.
   b. Have expert and novice users try the device and offer their criticisms.
3. Additional Features
   a. Extend the system to deal appropriately with multiple input signals.
   b. Research the feasibility of modifying the system to transfer data over a sub carrier frequency.
   c. Extend the iPhone application and MCU to send identification information, time of burial, and vital statistics of victims.

Demo-Day
Our plan for demo-day is to place a beacon outside the engineering building and use the iSlide device to locate it. We may also perform the experiment inside the engineering building if the devices function properly indoors.
Concerns

**Digital Signal Processing**
Since neither of us has a very strong background in signal processing, we anticipate that this will be the most challenging portion of our project. We have been able to find a significant amount of information on the topic, but this is essentially our first attempt at digital signal processing (DSP).

**Frequency Limitations**
Current standards governing the design of avalanche beacons require A1A modulation as described above. Without a sub carrier frequency data transmission will not be possible. Though this doesn't affect the core scope of our project, it would certainly prevent us from adding some of the additional features described above. Since we don't intend to commercialize iSlide, we aren't bound by ETSI standards, but in order for the device to be useful it must at least be compatible with other device currently in production. We're not yet sure if use of a different form of modulation will cause compatibility issues with other devices, but we have purchased an avalanche beacon for testing purposes and will be able to research the issue more deeply.

**Data Visualization**
Ideally, we'd like to implement a radar-type display on the iPhone GUI with the rescuer positioned in the middle of the screen and the location of all in-range transmitting beacons displayed relative to that point. However, due to the complexity of the near field flux pattern (Fig. 3) pinpointing the exact relative position of a transmitted signal on a radar-type will not likely be feasible. Our user display will probably be simplified to give a simple indication of the distance and direction a rescuer must follow along the flux lines to the location of the victim (see Fig. 1).

**Restricted Information on iPhone Hardware**
Apple allows certified manufacturers access to technical information regarding the iPhone's handshake and communication protocols. They don't, however, allow access to students. It is possible to "jailbreak" the device and gain root access [6] but documentation on hardware interfacing is rather sparse and jailbreaking the phone voids the warranty. The process can be reversed on a working phone but if the phone were to die while jailbroken, we would be out of luck.

We even went as far as registering a business with the State of Utah and submitting an application to Apple. We were promptly denied. In speaking with other iOS developers, it seems that, while Apple will take money from anyone who wants to develop iPhone applications, they are very selective about who they will allow to develop hardware. We will continue to research this and are considering the possibility of requesting that one of the current avalanche beacon manufacturers sponsor our research (even if in name only, without financial obligation). It's possible that a larger company could acquire the necessary permissions from Apple and that we would then perform our work under their license.
Radio Interference
Many electronic devices use 455 kHz as an intermediate frequency (IF). This is dangerously close to our 457 kHz target. It's possible that we may be dealing with interference from other devices, both in the lab and in the field. There's also documentation that suggests that GSM iPhones are particularly noisy for avalanche beacons. There are still a lot of unknowns for us here, since it is not clear exactly what causes the interference. Is it the IF? Is it the frequency of the processor? Is it the telephone functions of the phone or something about the GSM band? There may be simple solutions depending on the actual cause. The iPhone we're using, for example, is CDMA not GSM. We could also look into using an iPod touch instead. We've considered adding wire meshing to a silicon skin for the phone.

This will all require additional testing and research, but in the worst case we may just need a long cable separating iSlide from the iPhone. This wouldn't make for a very impressive device on demo-day, but would at least accomplish the core engineering objectives of our design.

Cost
Most avalanche transceivers are in the hundreds-of-dollars price range, as are iPhones and other smart phones. Ideally, we'd like to have a couple transceivers for testing and research as well as a couple of iPhones. We'll be using Jake's iPhone and we've purchased the antennae, the PodBreakout board, and an avalanche beacon, leaving us with a budget of about $300 for the project which significantly limits our options.

Conclusion
We've put a lot of thought and research into this project so far and we're excited about it! We feel confident that this project will give us an opportunity to demonstrate what we've learned over the course of our undergraduate studies and we look forward to the challenges before us. Wireless communications are so heavily used in our society today and seem to be the direction of technology in the future.
Appendix A - Gantt Chart
References


