University of Victoria
Department of Electrical and Computer Engineering
ELEC499A – Design Project

Tracker100

Technical Report

Submitted by Kyle Weston, Mike Blarowski
July 28, 2006

Table of Contents

List of Figures

0 Abstract

The purpose of this report is to describe the research and development of a Digital Avalanche Transceiver.

Travel and recreation in avalanche terrain has many inherent risks as most of the time it is done in a remote location, far away from any emergence response and rescue team. In the event of an avalanche and resulting burial, the only chance of survival is through "companion rescue," by a member of your own party. In an event of a full burial, when no visual signs exist, transceivers are the only proven effective way to locate a completely buried victim while he or she is still alive.

Decision has been made to design a transceiver operating at the standard 457 KHz frequency. This transceiver would have clear visual displays, incorporating distance to object indicator, direction indicator as well as an audible sound indicator. Minimum of two antennas would be used to precisely determine distance and direction to target. This would be achieved using microprocessor based system and transmitter/receiver modules. The goal was to design a device that meets or exceeds the currently available transceivers performance at a reduced cost.

AviTrans is a university project group consisting of Mike Blarowski, and Kyle Weston, both Electrical Engineering Undergraduate Students, supervised by Dr. Jens Bornemann. The time duration for this work was three months as per University of Victoria's ELEC 499A course guidelines.

The design of the Digital Avalanche Transceiver was a very rewarding experience. It has been demonstrated that a low power high performance transceiver can be built for a much reduced cost when compared to currently available units.

Due to the limited resources and time available for this project some of the features were not fully implemented and or tested, but the overall concept has been successfully demonstrated.

Opportunity for further transceiver improvements exists in several key areas such as antenna, filtering, firmware and power consumption. These improvements are described in detail in Section 7.

The working product demonstration and display proved to be a hit among the judges. Avitrans Tracker 100 claimed fist prize at the UVic Student design Competition on July 21st, 2006.

1 Introduction

AviTrans is a university project group consisting of Mike Blarowski, and Kyle Weston, both Electrical Engineering Undergraduate Students, supervised by Dr. Jens Bornemann. Our research focus is in developing a Digital Avalanche Transceiver to be used by enthusiasts and professionals alike in the backcountry. The time duration for this work is three months as per University of Victoria's ELEC 499A course guidelines.

Travel and recreation in avalanche terrain has many inherent risks due to the fact that most of the time it is done in a remote location , far away from any emergence response and rescue team. In the event of an avalanche and resulting burial, the only chance of survival is through "companion rescue," by a member of your own party. In an event of a full burial, when no visual signs exist, transceivers are the only proven effective way to locate a completely buried victim while he or she is still alive.

AviTrans decided to design a Digital Avalanche Transceiver operating at the standard 457 KHz frequency. This transceiver would have clear visual displays, incorporating distance to object indicator, direction indicator as well as an audible sound indicator. Minimum of two antennas would be used to precisely determine distance and direction to target. This would be achieved using microprocessor based system and transmitter/receiver modules. The goal was to design a device that meets or exceeds the currently available transceivers performance at a reduced cost.

With this in mind, please read on as we will describe the design process in detail.

2 Background Information

This portion of the report will focus on the goals of the project, both electrical and mechanical, and will act as base for the design and materials used to create the avalanche transceiver.

1.1 Small Size

In winter backcountry activates such as skiing or snowboarding the avalanche transceiver is to be worn by each individual at all times. Usually the skiers / snowboarders climb mountain on their own power so weight and size of the device is to be a small as possible.
A typical avalanche transceiver weighs approximately 200 grams, is approximately 10 cm x 15 cm in size with a thickness of about 2cm.

1.2 Ruggedness

Winter backcountry travel often involves cold temperatures and precipitation in the form of snow or rain in spring time. On descents or during a possible avalanche a great deal of vibration or shear stress can be applied to the transceiver.
Therefore the transceiver needs to be contained within a water and dust- proof casing which can be worn at all times. The unit needs to be easily removable during search and rescue operations. The materials used, need to withstand cold temperatures down to -30 degrees.

1.3 Economical

The average cost of currently available digital transceivers on the market, ranges between $350 and $450, which seems to be very high for a relatively simple device. The high costs of the transceivers, often prohibits their use and popularity especially by recreational users.

Our goal was to create a device that could be sold for not more than $200. The device was designed to have long lifetime, and operate on easily replaceable and commercially available batteries.

1.4 Power Efficient

In order to maximize the device operating time, we needed to address the power requirements of the transmitter and receiver. The entire system needed to be power efficient.

We decided to use 3 commercially available AAA batteries to keep the weight down and provide enough power to run the transceiver for 300 hours in TX mode and 150 hours in RX mode. Using 3 AAA batteries in series results in a 4.5 V supply. The battery capacity is 4500 mAh.

An additional component necessary to the design of a power efficient device is the addition of an LED to alert the user as to whether the device is on or off and how much power remains. This small addition can potentially save the user the cost of many batteries while needing a very small amount of current itself.

1.5 Compatibility

All currently available avalanche transceivers transmit and receive at 457 KHz with a +/- 100 Hz accuracy. The transmit protocol needs to conform to the ETS 300 718 standard [1] on pulse periods and pulse widths.

3 Hardware

3.1 Antennas

The antennas are one of the most important components of the avalanche transceiver design. Two orthogonally arranged antennas are to be used in the transceiver. These perpendicularly arranged antennas are used for determining the direction of the received signal source by isolating the electromagnetic flux-lines. One antenna is to be mounted on the front and the other one on the back side of the printed circuit board. This ensures adequate coverage, regardless of the transmitter's orientation.

Figure 1 illustrates the electromagnetic flux lines generated by antennas in this arrangement. At this low 457 kHz frequency, the antennas work according to the principle of a magnetic dipole.

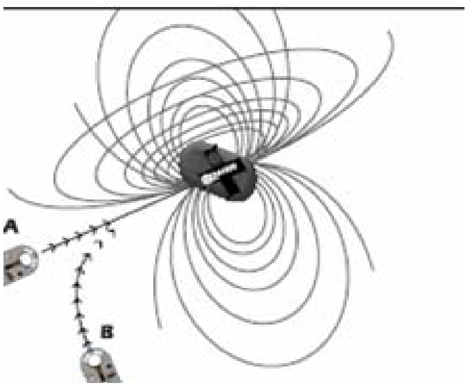


Figure 1 - electromagnetic flux pattern [2]

The required small size of the transceiver calls for small antenna dimensions. At this low operating frequency of 457 kHz it is physically impossible to design a small sized ¼ wavelength antenna, since the wavelength is over 600 meters.

The antenna size can be reduced greatly by insertion of a ferromagnetic material. Antennas are to be constructed by winding a wire around a ferrite rod. The antennas can be easily matched to the input circuit of the receiver / transmitter by means of a parallel resonant circuit. This circuit can be tuned to resonance at 457 kHz allowing for adjustment of noise or impedance matching.

The antennas are used by both the transmitter and receiver. The high Q of the ferrite material can result in a nicely tuned antenna with high selectivity which limits (filters) the unwanted interference signals outside of the 457 kHz band.

The voltage of the antenna input circuit, induced in the antenna coil by the electromagnetic field of the transmitter, is increased with the quality factor Q. It is assumed, that the input impedance of the receiver is high.

The terminal voltage of the ferrite rod antenna is determined by several parameters. It can be calculated as follows:

$$(1) \qquad U_A = Q * \frac{2\pi f_o}{\Delta f} * \mu_{eff} * n * A * E * \sin\alpha$$

where:
UA = terminal voltage
Q = circuit quality factor
Df = 3 dB bandwidth
μeff = effective permeability
n = number of turns
A = cross-sectional area of conductor loops
E = electrical field strength at the receiving position
α = angle of the magnetic field referred to the axis of the antenna

The above formula (1) shows, that:

- the electromagnetic field of the transmitter at the receiving position is important
- **directionality effect exists**
- **the induced voltage is higher with larger (thicker) ferrite rods**
- the ferromagnetic material has an influence
- **high numbers of turns lead to high antenna voltages**
- **a high quality factor increases the antenna voltage**

In addition, the **selectivity** of the antenna circuit increases with the increase in Q, which improves the **noise immunity** by reducing the bandwidth, thus rejecting the adjacent interfering frequencies.

The different parameters are partly dependent on each other. On one hand, a higher number of turns or an increased cross-sectional area results in a higher inductance, on the other hand - provided that the diameter of the wire is the same - it leads to a larger resistance. With an increasing number of turns the antenna voltage grows, but the increased resistance negates the raising Q factor (see formula 2).

Furthermore the effective permeability depends of the realisation of the winding, as the placement on the ferrite rod and the distance from the core. This means that the antenna design must be optimised for this application. All of the design details are outlined in the following sections. [3]

3.1.1 Resonant Resistance

The resonant resistance Rres can be calculated as follows:

(2) $$R_{RES} = \frac{1}{Q}\sqrt{\frac{L}{C}}$$

The quality factor Q of an existing antenna can be found using equation (3) by measuring the voltage at the Antenna terminals at the resonant frequency at both corner frequencies where the amplitude is 3 dB lower. The corner-frequencies define the 3 dB bandwidth (Δf).

(3) $$Q = \frac{f_o}{\Delta f}$$

3.1.2 LC values

To find the inductance and the capacitance values the following can be used for a resonant circuit:

(4) $$f_{res} = \frac{1}{2\pi\sqrt{LC}}$$

Choosing a practical value for the capacitance of 2.2 nF and Fres of 457 kHz, the required inductance is calculated using the following formula

(5) $$L = \frac{1}{4\pi^2 f_o^2 C} = \frac{1}{4\pi^2 (457kHz)(2.2nF)} = 55uH$$

3.1.3 Ferrite material selection

Next, a ferrite material needs to be selected in order to calculate the windings.
A good ferrite with a permeability (μ) of 2000 needs to be used. The temperature factor of the ferrite also has to be taken into account. This is not a big issue with low quality factors (~50) but with quality factors of 100 or more we have to take care, that with changing temperature the resonant frequency of our tuned antenna will not move, since the bandwidth of a high Q antenna is small compared with low Q-factors.

3.1.4 Effective Permeability

Next, we need now to determine the effective permeability (μeff) of the ferrite material. If it can not be obtained from the suppliers' data sheet, it can be determined by winding

100 turns onto the ferrite and then measuring the inductance. The following formula shows how to calculate the factor:
The inductance is calculated by:

(6) $$L = \mu_{eff} * w^2$$

where L=inductance
Ueff = effective permeability
w = number of windings

Calculating with 100 windings:

(7) $$L = \mu_{eff} * w^2 = \frac{L}{100^2} = \frac{L}{10^4}$$

To calculate the necessary windings we use:

(8) $$w = \sqrt{\frac{\mu_{eff}}{L}}$$

3.1.5 Wire selection and winding placement

The wire material and the placement of windings have an influence on the quality of the antenna. Commonly used wire diameters are in the 0,28 mm to 0,15 mm. As we can see from formula (2), reduction of the antenna capacitor leads to a higher necessary inductance which will increase the effective antenna voltage and also the resonant resistance.

The higher necessary windings on the ferrite lead to more losses due to spread capacitances between the windings and therefore, a reduction of the quality factor Q.

If high number of turns is necessary, reducing of the thickness of the wires isolation can be taken in account. Same number of windings and diameter of the wire can lead to different Q factors if the isolation and the accuracy of the windings are changed.

3.1.6 Antenna Tuning

The antenna has to be tuned to the desired frequency, since the ferrite itself and the capacitor have some tolerances. This is achieved with an adjustable trimmer capacitor placed as close as possible to the windings. A phase-gain analyser would be the ideal tool to quickly find the resonant point.

The above steps describe the design process for an antenna best suited for our application. Due to time constraints and lack of the necessary parts and equipment, the working prototype unit as seen in Appendix C, takes advantage of a standard 45 MHz receiver demo board from Philips Semiconductors. Since a much higher frequency was used, a single whip style antenna was used to validate the transceiver design.

As an improvement of this design a third virtual antenna could be created by means of a differential front end amplifier to which the two physical antennas are connected. The virtual antennas signal magnitude would be derived from the phase information generated by the differential amplifier.

Another variation would be to add a third physical antenna, which could be turned on when needed to steer (narrow) the beam pattern of the two receiving antennas. This technique is often employed in radio direction finding applications as described the ARRL handbook [4].

3.2 Receiver

Superhetrodyne receiver configuration was chosen for this design. Philips SA606 mixer FM IF IC has been selected due to its versatility, low-power consumption and small size. This IC is the main building block of the receiver circuit as seen in the receiver schematic enclosed in Appendix A5).

It incorporates a mixer/oscillator, two limiting intermediate frequency amplifiers, quadrature detector, logarithmic received signal strength indicator (RSSI), voltage regulator and audio and RSSI op amps all of which are shown in figure below

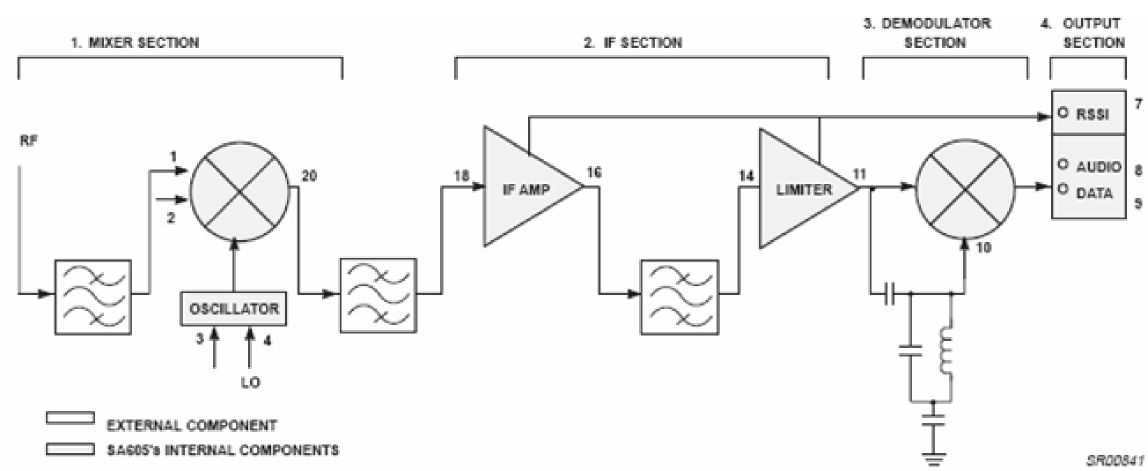


Figure 2 – SA606 Receiver IC [5]

3.2.1 Mixer Section

The Receiver mixes the local oscillator frequency which was selected to be 912 kHz with the incoming RF signal of 457 kHz using the on board gilbert-cell mixer.
In the mixer stage of the receiver, the local oscillator signal multiplies with the incoming signal, producing sum and difference (beat frequencies), both above and below the incoming signal as per equations (9) and (10).

(9) $F_2 = F_{LO} + F_{IN} = 912kHz + 457kHz + = 1369kHz$

(10) $F_2 = F_{LO} - F_{IN} = 912kHz - 457kHz + = 455kHz$

The lower frequency was selected to be the IF stage. In order to filer (reject) the higher frequency produced by the mixer, a Murata SFG455 455 kHz ceramic band-pass filer was used. With a 30 kHz bandwidth effectively filters out the high frequency.

3.2.2 Limiter Section

The filtered IF signal is then amplified by the fist stage IF and then filtered again by another band-pass filer identical to the one used at the output of the mixer, the signal then passes through a limiting amplifier.

3.2.3 Demodulator Section

The demodulator section of the IC is not used as we're not demodulating information but simply detecting the incoming signal strength.

3.2.5 Output Section

Once the signal goes through the second stage IF amp, the receiver signal strength indicator (RSSI) signal is produced. The RSSI signal indicator is an integral part of this design. It is used by the microprocessor to determine incoming signal direction and the distance from the transmitting transceiver to the receiving one. This process is discussed in detail in the firmware (section 4) of this document. The complete receiver schematic is attached in Appendix A-#?

The project time-line did not allow for the proper layout and fabrication of the receiver operating at 457 kHz. Instead, to validate the design a standard Philips SA606 Receiver demo board was used. It was tuned to work with a 45 MHz incoming signal and many attempts were made to match the front end to work with the desired 457 kHz frequency. Unfortunately due to the limited availability of surface mount inductors and capacitors in our lab facility we were not able to achieve high enough dynamic range and a decision was made to use the demo board at the standard 45 MHz frequency to validate the transceiver design.

3.3 Transmitter

It has proven to be very difficult to find a transmitter module which works at 457 kHz. Therefore a discrete transmitter had to be designed.

The first challenge was to find an off the shelf 457 kHz crystal which, as it turns out doesn't exist.. Since the quarts crystal could not be sources, it was decided to use an Epson SG8002 programmable oscillator module running at 4.57 MHz. The Epson offered an inexpensive way to get a custom crystal and the oscillator in one package.
To generate the required 457 kHz carrier frequency, a 74HC4017 decade counter/divider was used to a divide the 4.57 MHz signal by 10 generating the carrier. The oscillator itself is a bit noisy but when divided down to the low frequency region the phase noise is very low.

The carrier signal then drives a complementary pair final amplifier which boosts the signal to the required level. The transmitter can be keyed by the microcontroller through the enable pin (#) on the Epson oscillator.

The final portion of the circuit is a lowpass filter which gets rid of unwanted high frequency harmonics. Complete schematic of the transmitter is attached in Appendix ##

Since a standard 45 MHz demo board was used for the receiver (Section 3.2), to validate the design, a high frequency signal generator was used to provide the 45 MHz carrier needed for a complete transceiver demo.

3.4 Signal Filters

As described in Section 3.2.1, frequency of 455 kHz was selected for the IF stage. In order to reject the higher image frequency of 1369 kHz produced by the mixer, two Murata (SFG455) 455 kHz ceramic band-pass filers were used. First filter is located between the mixer and the IF amplifier and the second filter located between the IF amplifier and the limiter further reduces the unwanted frequencies. With a 30 kHz bandwidth effectively filters out the high frequency. Appendix A-5 shows the locations of these filters in the circuit.

To further improve the overall system a front end filter should placed between the receive antennas and the RF input of the SA606 receiver module.

3.5 Microcontroller

An Atmel ATMEGA16 was selected as the microcontroller of choice, allowing for 32 I/O pins and 16 Kbytes of flash memory. Programming was accomplished via the on-chip JTAG interface. The firmware controls the operation of the LED displays, the speaker, the battery voltage sensor, analysis of direction and distance from the receive antennas as well as other functions such as receive and transmit mode switching.

3.6 Display / LED's / Speaker

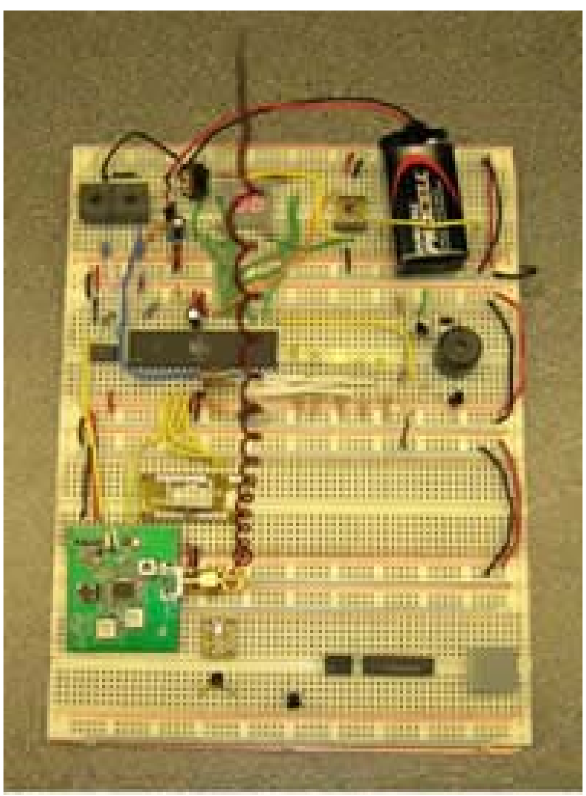
Dual 8 segment LED display was used to display battery power at power-up and distance readings during the search mode operation.

Several LED's mounted in a semi circle are to be used to indicate received signals direction. Both the display and the LED's are controlled by the ATMEL MCU and are based on the receiver signal strength indicator (RSSI).

A Star Micronics speaker, model HMB-01, was selected to complement the visual indicators. This self-driven speaker is controlled by the MCU and the RSSI level. The pitch of the sound varies in sync with the RSSI level to indicate the approximate distance of a target. The pitch increases with the decreasing distance to target.

3.7 PCB Layout and Fabrication

The original plan included the layout, fabrication and assembly of a fully functional circuit board, but due to the limited time and resources, the system was prototyped using a bread-board and the Philips SA606 receiver front end demo board. The following photo shows the assembled system which was used during the poster presentation demo.



4 Firmware

4.1 Transmit Protocol

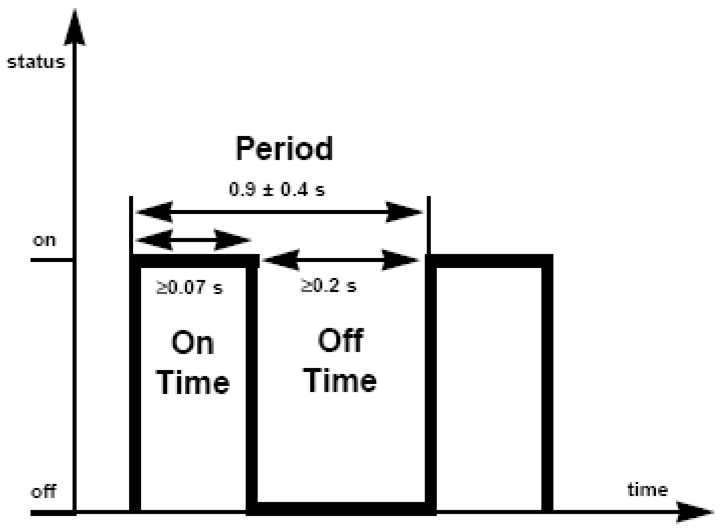


Figure 3 - ETS 300 718 standard on pulse periods and pulse widths

The transceiver transmits with a pulse repetition rate (PRR) of 1 second, with an ON time of 70 ms. This falls within the specification in the ETS 300 718 standard [2] which allows for > 70 ms of ON time as seen Figure 3. An obvious benefit to having a low duty cycle is that the power consumption is reduced preserving the battery life, however having a duty cycle that is too low could mean that some of the older analog transceivers will not be able to detect the signal.

The transceiver powers up in transmit mode which is the standard mode of operation. It can be switched to receive mode by holding down the transmit/receive button for a period of at least two seconds. This is to ensure that the transceiver can not be switched to receive mode accidentally, which could prove to be fatal if the person where to be caught in an avalanche. While in transmit mode the transceiver will transmit pulses at the rate specified, consuming as little power as possible.

4.2 Receive Protocol

The antenna signal strength is read approximately every 5 ms from the RSSI indicator of the Philips SA606 IC when the transceiver is in the receive mode. With two antennas the active receiving antenna would be switched after every interval. In an effort to reduce noise each signal is averaged over 100 samples. The working prototype used one antenna, as due to time constraints the antenna multiplexing circuit was not completed.

Since in an outdoor environment there is some degree of ambient noise present, the noise level should be sampled periodically so that it does not get mistaken for the signal. This was not implemented in the current firmware version but could possible be implemented in a future release version.

4.3 Distance Analysis

The distance is to be calculated from the two receive antennas using the norm for Euclidian space

$$\text{distance} = \sqrt{\mathrm{T}(\mathrm{V}_\mathrm{A})^2 + T(V_B)^2}$$

where V_A and V_B represent the voltage measured from the RSSI output of the SA606 for antenna A and B respectively. T( ) represents a transformation from voltage to distance measured in feet.

This assumes that receive antennas are arranged orthogonally, constituting orthogonal vectors. This is expected to be only a rough estimate of the true distance since the near field pattern is highly irregular, especially at close distances as can be seen in **Error! Reference source not found.**. Furthermore the RSSI signal is also dependent on the antenna orientation as well as the battery life (output power) of the transmitter.

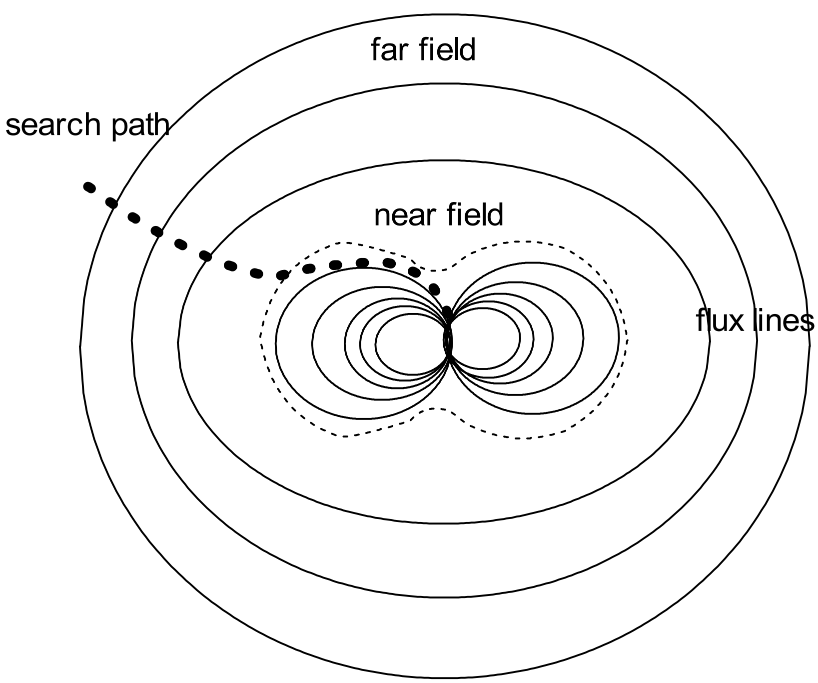


Figure 4 - Near field/Far field concept for search techniques

Since the working range is within ¼ the wavelength of the antenna at 457 kHz the it is expected that the received power obeys the Friis equation [6] as given by

$$P_{Rx} = P_{Tx}\frac{G_{Tx}G_{Rx}\lambda^2}{16\pi^2 d^2 L}$$

where

G_{Tx} = transmitter antenna gain

G_{Rx} = receiver antenna gain

λ = wavelength

d = distance between transmitter and receiver

L = system loss factor (≥ 1)

Solving for d and grouping the unknowns together we have

$$d = \frac{C\lambda^2}{P_{Rx}^{\ 1/2}}$$

The distance was then plotted versus voltage for a number of data points in an indoor environment and the plot shown in Figure 6 was obtained. It can be seen from the equation for the regression line that the distance to target varies roughly as the inverse square root of the voltage, which is what was expected.

It was observed that the distance reading was greatly affected by the proximity of walls, metal objects and people standing nearby. This is due to the frequency used.
The SA606 receiver demo board used operates at 45 MHz where the wavelength is 6.6 meters the human body makes an ideal ¼ wave antenna.

These factors would not affect the distance readings as much if 457 kHz was used as the carrier frequency as intended since at that frequency the wavelength is 656 meters and the skin depth is much greater. Therefore the signal can penetrate walls much more easily and is not affected by metal object, snow etc. This was the main reason why the 457kHz frequency was originally selected as the standard operating frequency for avalanche transceivers as described in ETS 300 718 standard [1].

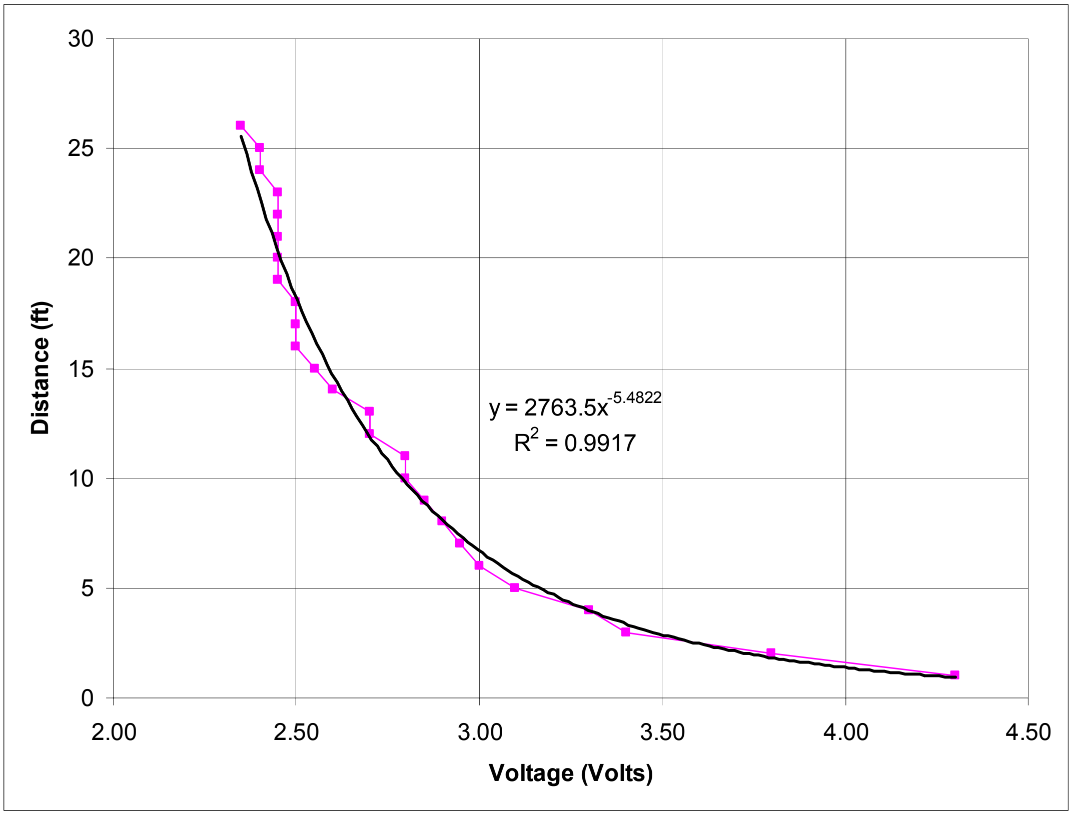


Figure 5 - Plot used for the voltage to distance calculation

4.4 Direction Analysis

Since the two receive antennas are orthogonal the direction can be calculated using the simple trigonometric relationship

$$\theta = \arctan A / B$$

,where A and B represent the received signal strength from antenna A and antenna B respectively. However, as can be seen from Figure 6 below, for a given angle θ there exist four possibilities for the true direction to the target. This is because the receive antenna has equal frontal and rear lobes. If a more directional (narrower) antenna beam pattern could be achieved, possible by adding a third antenna to form the beam pattern, the problem of direction ambiguity could be solved.
However the probability of finding the signal is reduced. Ideally one would want to have an omni-directional antenna for transmitting and a directional antenna for receiving. Due to the lack of time and resources for this project, one antenna was used for the prototype. As a result the direction functionality was not tested. A solution for selecting the true target location from four possible target locations needs to be implemented.

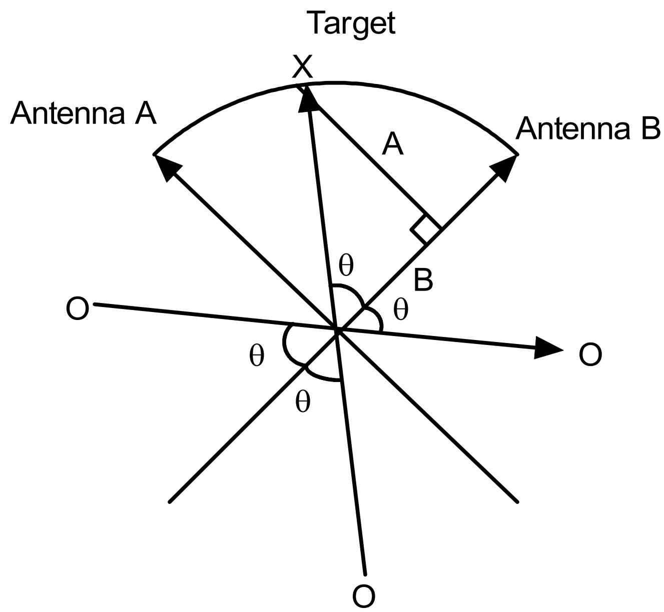


Figure 6 - Direction analysis diagram. X represents the true target location and O represents false target locations.

4.5 Direction Display

The direction display consists of five LED's oriented in a row, which are connected directly to the microcontroller through current limiting resistors. Each LED corresponds to a direction with 18 degree (90/5) accuracy. An LED will be lit when the direction corresponding to its range is detected.
Since the prototype used a single antenna, the LED's arranged in a single row, were used to indicate the signal strength during the demo.

4.6 Distance Display

The distance to target is displayed in feet on the dual 7-segment display which is connected directly to the microcontroller and multiplexed to save power. Each digit is pulsed with a duty cycle of 25% to give a maximum current consumption of 35 mA.
An LED display was chosen over an LCD display due to its ability to operate at very low temperatures often encountered in normal usage conditions. Most LCD displays do not perform well below -10 degrees Celsius, hence would be unsuitable for an avalanche transceiver. Under normal conditions, the 7-segment display would not be active in transmit mode to save power.

4.7 Speaker Driver

The speaker used is a Star Micronics HMB-01 and has self-contained driving circuitry. The drive frequency for the speaker is generated from TIMER2 in Compare to capture mode. This way, all that is necessary to do in software, is to load the output compare register OCR2 with the appropriate value and each time the timer reaches the value in the Compare register the output pin connected the speaker is toggled. The timer runs continuously and is reset to zero when an overflow is detected. The frequency range used is from 1.5 kHz to 3 kHz with the higher frequencies corresponding to smaller distances between transmitter and receiver. See Appendix B-1 for flow of control. The following formula is used for OCR2 register value

$$OCR2 = \frac{f_{clk}}{2N\, f_{OC2}} - 1$$

$$\textit{where}$$

$$f_{clk} = \text{System clock frequency}$$

$$N = \text{clock prescalar value}$$

$$f_{OC2} = \text{target frequency}$$

Although the microcontroller outputs a square wave to the speaker, the internal driving circuitry of the speaker filters it to appear sinusoidal resulting producing a desirable sound quality.

5 Cost Analysis

One of main design goals and a major driving force behind this project was the overall cost of the transceiver. Table 1 below shows the approximate costs of the main components of the avalanche transceiver based on low volume production.

Component	Cost
Microcontroller	$ 8.00
Oscillators	$ 15.00
PCB	$ 10.00
Enclosure	$ 50.00
Ferrite (for Antennas)	$ 5.00
Misc. Components	$ 10.00
Total Cost	**$ 98.00**

Table 1: Cost Analysis

As it can be seen from the above table the product can be built for less than a $100. With a good product acceptance and demand the units can be built in volume at a price closer to $50 range. Assuming a $100 cost, the finished product could retail in the range of $200 to $250 which is about half of the average cost of currently available digital transceivers on the market which range between $350 and $450.

6 Conclusions

The design of the Digital Avalanche Transceiver was a very rewarding experience. It has been demonstrated that a low power high performance transceiver can be built for a much reduced cost when compared to currently available units.

Due to the limited resources and time available for this project some of the features were not fully implemented and or tested, but the overall concept has been successfully demonstrated. The working product demonstration and display proved to be a hit among the judges. Avitrans Tracker 100 claimed fist prize at the UVic Student design Competition on July 21st.

There are several possible areas of improvement all of which are outlined in the following Recommendations (Section 7).

7 Recommendations

Opportunity for transceiver design improvements exists in several key areas of the design.

Antenna:
A third virtual antenna could be created by means of a differential front end amplifier to which the two physical antennas are connected. The virtual antennas signal magnitude would be derived from the phase information generated by the differential amplifier.

Another possible variation would add a third physical antenna. When activated, this antenna would steer (narrow) the beam pattern of the two receiving antennas helping resolve the directional ambiguity as described in section 4.4.

Filtering:
To further improve the overall system a front end filter should be placed between the receive antennas and the RF input of the SA606 receiver module.

Firmware:
The ambient noise level present in the outdoor environment should be sampled periodically so that it does not get mistaken for the real signal.

Power Consumption:
To minimize power consumption, the current limiting resistors for the LEDs can be increased to where the LEDs are just visible in ambient light. The LEDs and the microcontroller are the main source of current consumption. Also the microcontroller should be put in sleep mode when not transmitting or sampling data.

8 References

[1] European Radio communications Office. 1999: "Report Form for Testing to ETS 300 718" http://www.ero.dk/documentation/docs/doc98/official/pdf/RP300718.PDF

[2] B. Edgerly, J.Hereford. 1998: Digital Transceiving Systems: The next generation of avalanche beacons.

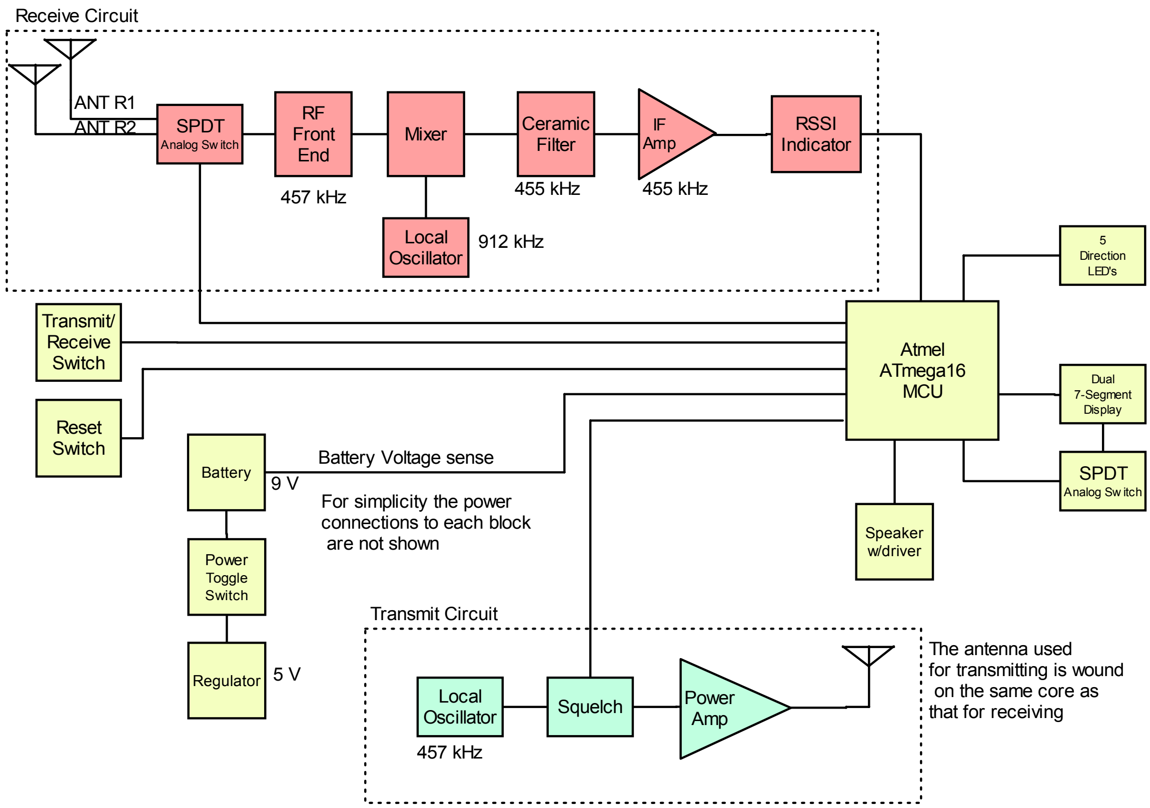
[3] Author, Year, publisher, Antennas and Propagation: p

[4] ARRL Handbook for Radio Communications, 2005: p.13.18-13.19

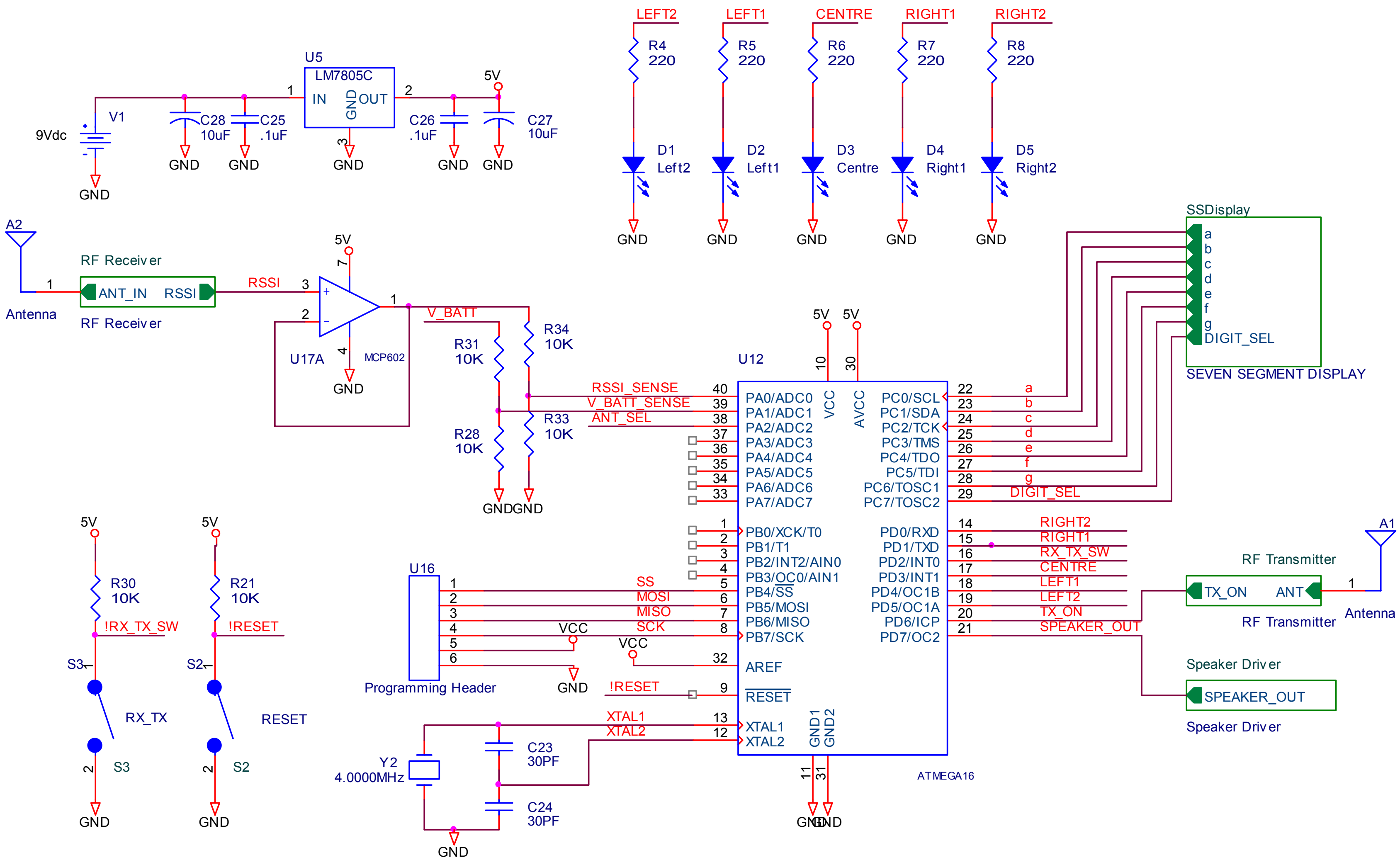
[5] Phillips Semiconductors, AN1994: "Reviewing key areas when designing with SA605".

[6] 2006: "Free Space Path Loss," http://www.rfcafe.com/references/electrical/path_loss.htm

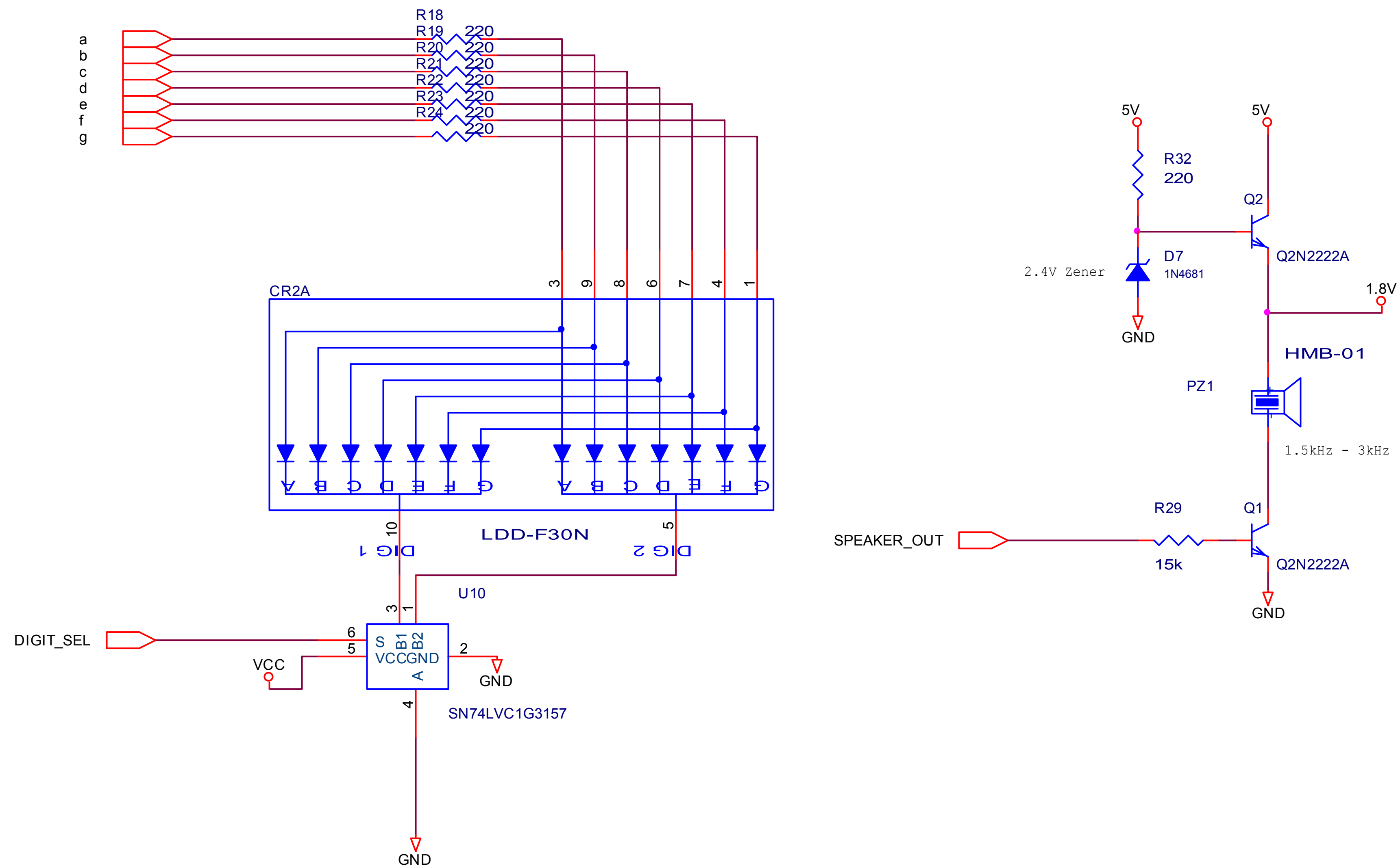
Appendix A-1 System Block Diagram



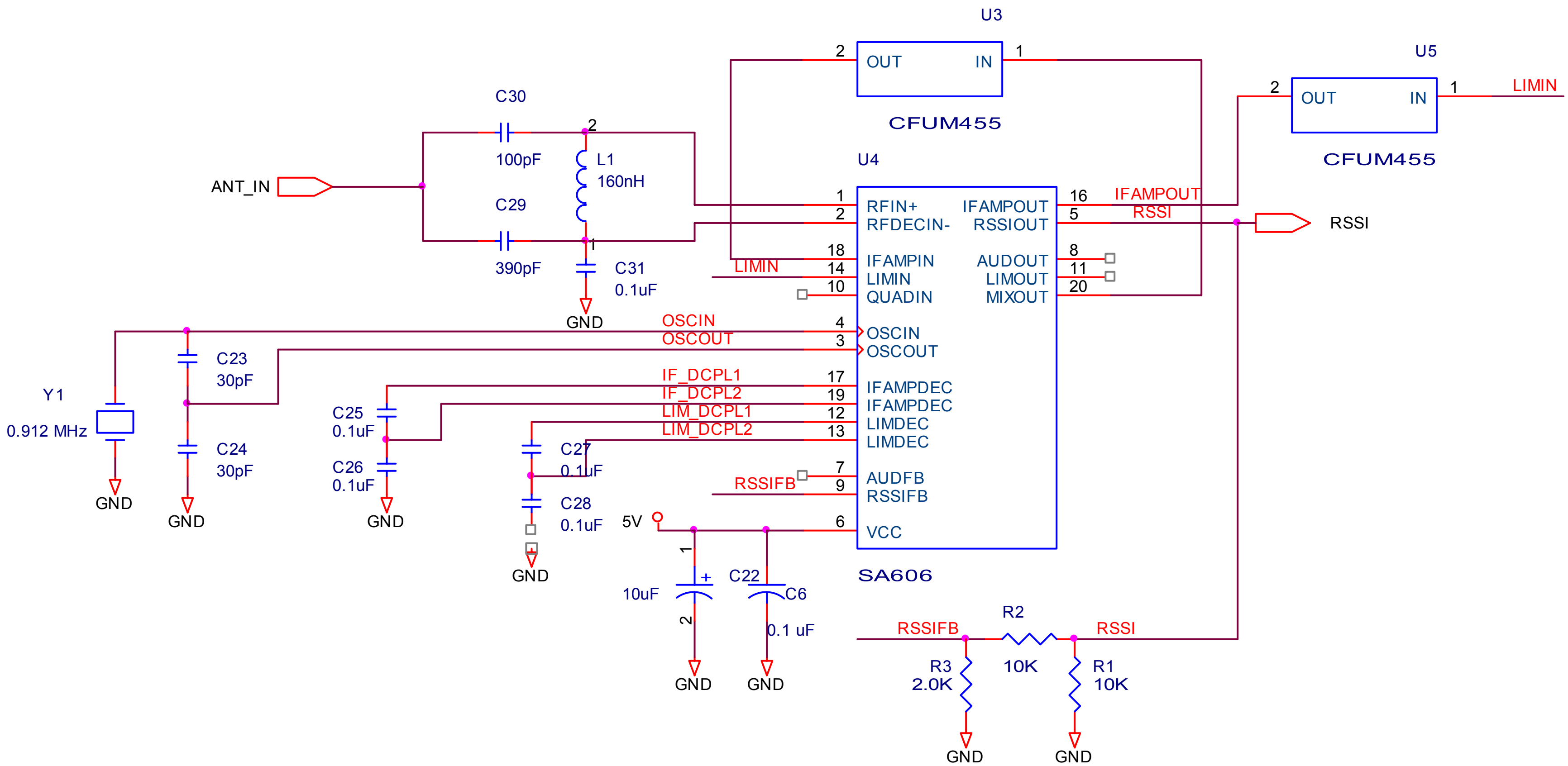
A-2 Microcontroller.sch



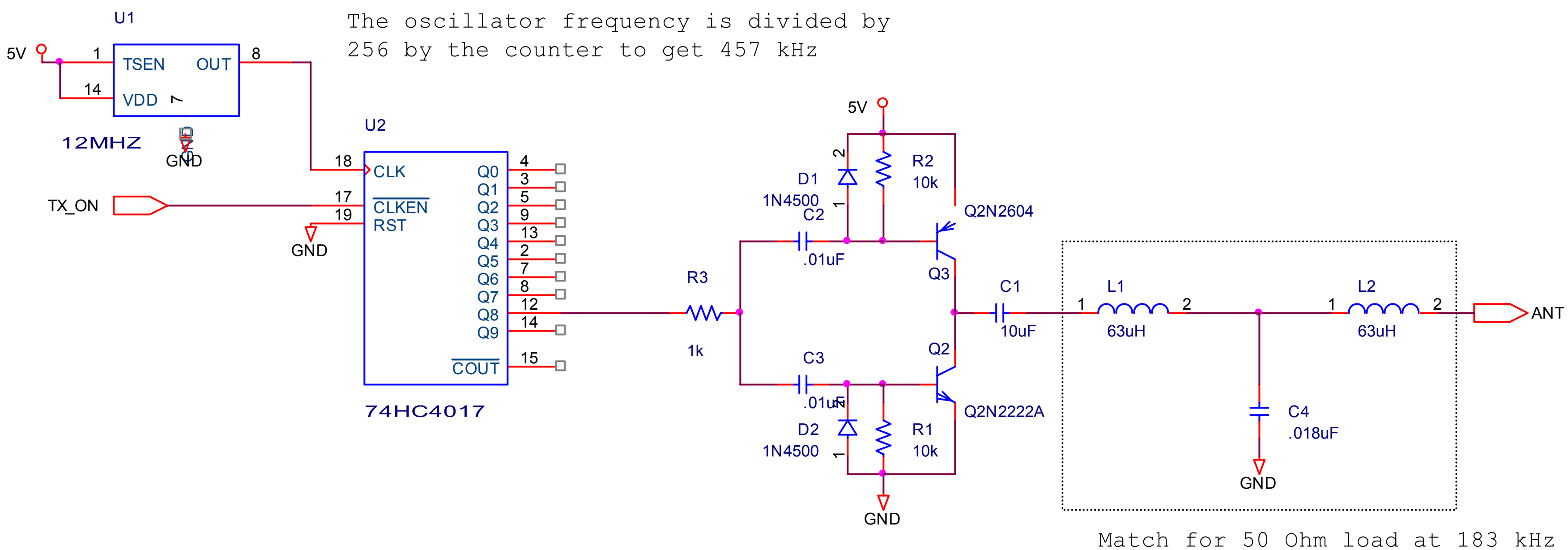
A-3 Seven-Segment Display.sch, Speaker Driver.sch



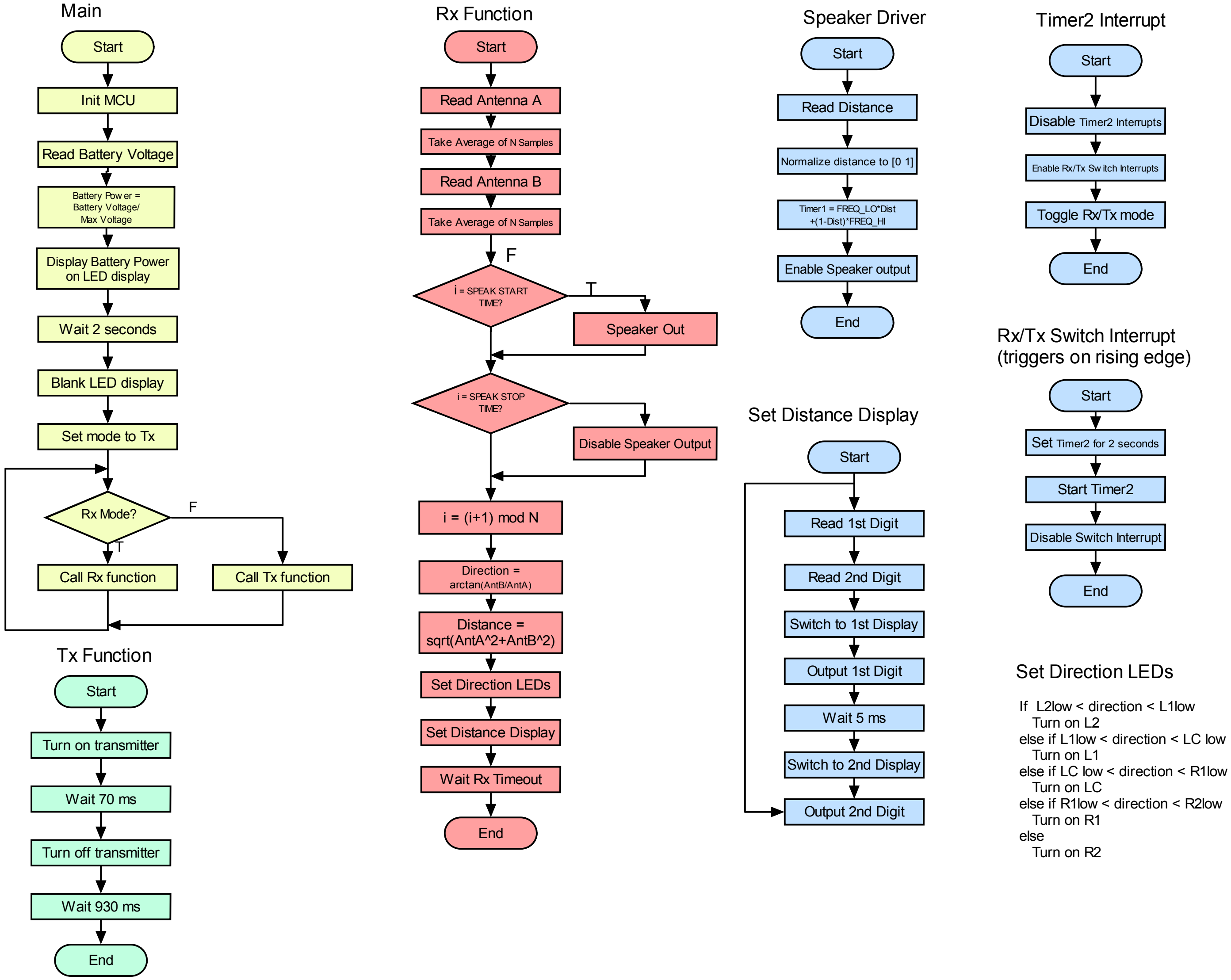
A-5 RF Receiver.sch



A-6 RF Transmitter.sch



Appendix B-1 Firmware Flowchart



B-2 Code Listing

```
/***************************************************************************
*		File: avitrans.c
*		Author:	Kyle M Weston
*		Creation Date:		June 26 2006
*
*		Description:		This is the firmware controls all the functionality of
*		a digital avalanche transceiver, including distance and direction
*		analysis from to receiving antennas.  It also controls the frequency of
*		an attached speaker to indicate the distance to the target and displays
*		the distance in ft on a dual LED display
*
***************************************************************************/

#include <inttypes.h>
#include <avr/io.h>
#include <avr/interrupt.h>
#include <avr/sleep.h>
#include <math.h>
#include <util/delay.h>

#include "iocompat.h"

/* Definitions */
#define F_CPU 4000000UL

#define TRUE 1
#define FALSE 0
#define MAX_BATTERY_VOLTAGE 5.0
#define VREF 5.0
#define SWITCH_WAIT_TIME 2
#define TX_ON_TIME 70
#define RX_INTERVAL 50
#define SPEAKER_FREQ_LOW 1500
#define SPEAKER_FREQ_HIGH 2800
#define SPEAKER_ON_TIME 0.25
#define SPEAKER_INTERVAL 2
#define MAX_DISTANCE 50
#define RSSI_SLOPE 18.9473
#define RSSI_INTCP -121.263

#define RSSI PA0
#define BATTV_P PA1
#define ANT_SEL PA2

#define SSD_A PC0
#define SSD_B PC1
#define SSD_C PC2
#define SSD_D PC3
#define SSD_E PC4
#define SSD_F PC5
#define SSD_G PC6
#define DIGIT_SEL PC7

#define RIGHT2 PD0
#define RIGHT1 PD1
#define RX_TX_SW PD2
#define CENTER PD3
#define LEFT1 PD4
#define LEFT2 PD5
#define TX_ON PD6
#define SPEAKER_OUT PD7


/* Macros */
#define BITREAD(sfr,bit) ((sfr & _BV(bit)) >> _BV(bit))
```

```
#define BITCLEAR(sfr,bit) (sfr &= ~_BV(bit))
#define BITSET(sfr,bit) (sfr |= _BV(bit))
#define BIT_IS_SET(sfr,bit) ((sfr & _BV(bit)) == _BV(bit))
#define BIT_IS_CLEAR(sfr,bit) ((sfr & _BV(bit)) != _BV(bit))


/* function prototypes */
void do_tx_routine(void);
void do_rx_routine(void);
void set_dir_leds(int dir);
void setDDDisplay(int ad1,int ad2);
void sound_out(double dist);
uint8_t conv_ssd(uint8_t a_digit);
void wait_ms(uint16_t time_ms);
uint16_t read_adc(uint8_t channel);

/* global variables */
uint8_t rxmode = FALSE;
uint8_t gd1 = 0;
uint8_t gd2 = 0;


ISR (TIMER0_COMP_vect)
{
        static uint8_t i = 0;

        switch(i)
        {
                case 0:
                        PORTC = conv_ssd(gd1);          //output 1st digit
                        BITSET(PORTC,DIGIT_SEL);
                break;
                case 4:
                        PORTC = conv_ssd(gd2);          //output 2nd digit
                        BITCLEAR(PORTC,DIGIT_SEL);
                break;
                case 1:
                case 2:
                case 3:
                case 5:
                case 6:
                case 7:
                        PORTC &= 0x80;                          //blank the display to save power
                break;
        }
        i = (i+1)%8;
}

ISR (TIMER1_OVF_vect)
{
  //Toggle Rx/Tx mode if button is held for SWITCH_WAIT_TIME
        if(BIT_IS_CLEAR(PIND,RX_TX_SW))
        {
                rxmode ^= 1;
                if(rxmode == 0)
                {
                        //display 'tr' for transmit
                        setDDDisplay('t','r');
                        //clear all LEDs
                        PORTD &= ~(_BV(RIGHT2) | _BV(RIGHT1) | _BV(CENTER) | _BV(LEFT1) | _BV(LEFT2));
                        //turn speaker off
                        BITCLEAR(TCCR2,COM20);
                }
                else
                {
                        //display 're' for transmit
                        setDDDisplay('r','e');
                }
        }
        //disable timer 1 overflow interrupts
```

```
        BITCLEAR(TIMSK,TOIE1);
        //clear any pending INT0 interrupts
        BITSET(GIFR,INTF0);
        //re-enable INT0 interrupts
        BITSET(GICR, INT0);

}

ISR (INT0_vect)
{
        // Enable timer 1 overflow interrupt
        //Preset timer for SWITCH_WAIT_TIME
        //disable any further interrupts until timer expires
        BITCLEAR(GICR, INT0);
        BITSET(TIMSK,TOIE1);
        TCNT1 = (65536 - SWITCH_WAIT_TIME*F_CPU/256);
}

void ioinit (void)
{
  //Setup direction of I/O ports
        DDRA = 0x04;
        DDRB = 0x00;
        DDRC = 0xFF;
        DDRD = 0xFB;

        //Setup INT0 to generate an interrupt on a falling edge
        //This will be used for the rx/tx switch
        MCUCR = _BV(ISC01);
        GICR = _BV(INT0);

        //Setup Timer 0 for 8 bit CTC mode, Prescale clock by 1024
        //This will be used to multiplex the 7 segment display
  TCCR0 = _BV(WGM01) | _BV(CS01) | _BV(CS00);
        BITSET(TIMSK,OCIE0);        //enable Timer 0 interrupts
        OCR0 = 0x6; //Set display interval to 10ms

        //Setup Timer 1 for 16 bit normal operation. Prescale clock by 256
        //This will be used to time the rx/tx switch
  TCCR1A = 0x00;
  TCCR1B = _BV(CS12);

        //Setup Timer 2 for CTC mode, Prescale clock by 8
        //This will be used to drive the speaker
        TCCR2 |= _BV(WGM21) | _BV(CS21);

        //Setup ADC for single conversion
        ADCSRA = _BV(ADPS2) | _BV(ADPS0);
        ADMUX |= 0x1;       //Configure PA1 as an ADC channel

}

int main (void)
{
        double battv = 0;
        uint16_t adc_result = 0;

  ioinit ();
        sei ();

  //Read Battery Voltage
        adc_result = read_adc(BATTV_P);
        battv = 2*adc_result*VREF/1024;
        //display battery voltage
        setDDDisplay((int)battv, (int)(battv*10)%10);
        wait_ms(1000);
        //display 'tr' for transmit

        setDDDisplay('t','r');
```

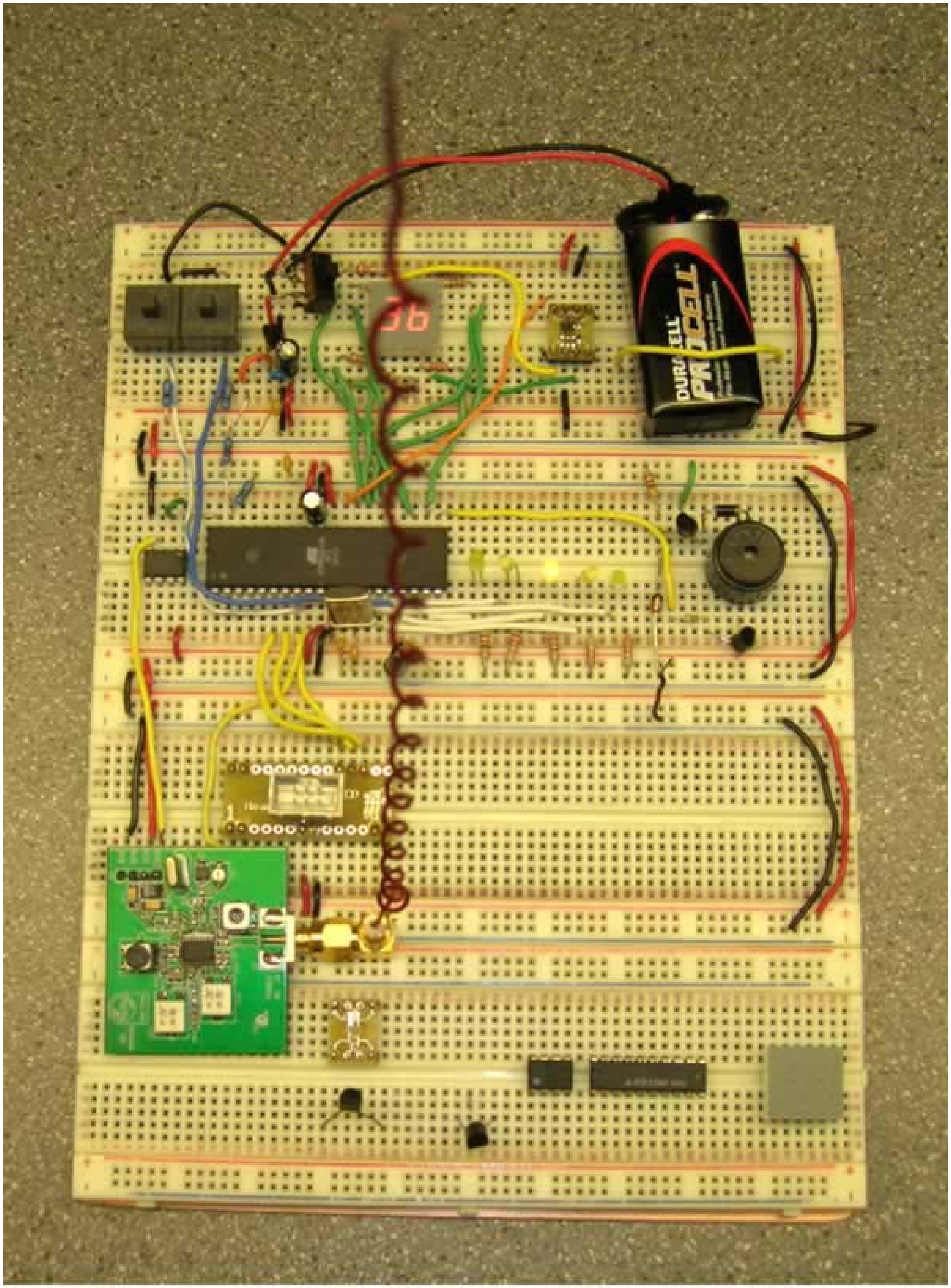
```
        wait_ms(1000);

        //Setup ADC for free running conversion on RSSI pin
        SFIOR = 0x00;
        ADCSRA = _BV(ADEN) | _BV(ADATE) | _BV(ADPS2) | _BV(ADPS0);
        ADMUX = 0x00;       //Configure PA0 as ADC channel

        //rxmode = 1;

  while(1)
        {
                if(rxmode)
                {
                        do_rx_routine();
                }
                else
                {
                        do_tx_routine();
                }
        }

  return (0);
}



void do_tx_routine()
{
        BITSET(PORTD,TX_ON);        //Turn on transmitter
        wait_ms(TX_ON_TIME);
        BITCLEAR(PORTD,TX_ON);  //Turn off transmitter
        wait_ms(1000-TX_ON_TIME);
}

#define N 100
void do_rx_routine()
{
        static uint8_t i = 0;
        uint8_t k;
        double v_ant, dir, dist;
        uint16_t adc_result, adc_sum;

        //switch to antenna A
        //BITCLEAR(PORTA,ANT_SEL);

        //Read RSSI Voltage, take an average of N samples
        adc_sum = 0;
        for(k = 0; k < N; k++)
        {
                adc_sum += read_adc(RSSI);
        }
        adc_result = adc_sum/N;
        v_ant = 2*adc_result*VREF/1024;
        dist = 2763.5*pow(v_ant,-5.4822);
        if(dist > 99)
        {
                dist = 99;
        }
        if(dist < 0)
        {
                dist = 0;
        }
        //display distance
        setDDDisplay((int)(dist)/10, (int)(dist)%10);
        set_dir_leds((int)dist*(90/MAX_DISTANCE));

        //display RSSI voltage
        //setDDDisplay((int)(dist)/10, (int)(dist)%10);
        //set_dir_leds((int)dist*(90/30));
```

```
        if(i == 10)
        {
                sound_out((MAX_BATTERY_VOLTAGE - v_ant)*10);
        }
        if(i == 11)
        {
                i = 0;
                BITCLEAR(TCCR2,COM20);
        }
        i++;

        wait_ms(RX_INTERVAL);

}

uint8_t conv_ssd(uint8_t a_digit)
{
        uint8_t ssd_out = 0;

        switch(a_digit)
        {
                case 0:
                        ssd_out = 0x3f;
                break;
                case 1:
                        ssd_out = 0x06;
                break;
                case 2:
                        ssd_out = 0x5b;
                break;
                case 3:
                        ssd_out = 0x4f;
                break;
                case 4:
                        ssd_out = 0x66;
                break;
                case 5:
                        ssd_out = 0x6d;
                break;
                case 6:
                        ssd_out = 0x7c;
                break;
                case 7:
                        ssd_out = 0x07;
                break;
                case 8:
                        ssd_out = 0x7f;
                break;
                case 9:
                        ssd_out = 0x67;
                break;
                case 't':
                        ssd_out = 0x78;
                break;
                case 'r':
                        ssd_out = 0x50;
                break;
                case 'e':
                        ssd_out = 0x79;
                break;
                default:
                        ssd_out = 0xff;
        }
        return ssd_out;
}

void set_dir_leds(int dir)
{
```

```
            //clear all LEDs
            PORTD &= ~(_BV(RIGHT2) | _BV(RIGHT1) | _BV(CENTER) | _BV(LEFT1) | _BV(LEFT2));

            if(dir >= 0 && dir < 18)
            {
                        BITSET(PORTD,LEFT2);
            }
            else if( dir >= 18 && dir < 36)
            {
                        BITSET(PORTD,LEFT1);
            }
            else if( dir >= 36 && dir < 54)
            {
                        BITSET(PORTD,CENTER);
            }
            else if( dir >= 54 && dir < 72)
            {
                        BITSET(PORTD,RIGHT1);
            }
            else
            {
                        BITSET(PORTD,RIGHT2);
            }

}

void setDDDisplay(int ad1,int ad2)
{
            gd1 = ad1;
            gd2 = ad2;
}

void sound_out(double dist)
{
            double alpha = 0;
            BITSET(TCCR2,COM20);        //enable speaker output (OC2)
            //normalize the distance to the interval [0,1]
            alpha = dist/MAX_DISTANCE;
            if(alpha > 1)
            {
                        alpha = 1;
            }
            if(alpha < 0)
            {
                        alpha = 0;
            }

            OCR2 = (int)((F_CPU/(2*8*SPEAKER_FREQ_LOW))*alpha + (1-alpha)*(89));
//F_CPU/(2*8*SPEAKER_FREQ_HIGH) = 83

            //allow speaker to sound for SPEAKER_ON_TIME
            BITSET(TIMSK,TOIE0);
}

uint16_t read_adc(uint8_t channel)
{
            uint16_t adc_result;

            ADMUX = channel;
            //enable ADC driver
            BITSET(ADCSRA,ADEN);
            BITSET(ADCSRA,ADSC);
            while(BIT_IS_CLEAR(ADCSRA,ADIF));
            adc_result = ADCL;
            adc_result |= (ADCH << 8);
            //disable ADC driver
            BITCLEAR(ADCSRA,ADEN);
            return adc_result;
}
```

```
void wait_ms(uint16_t time_ms)
{
	uint16_t t;

	for(t = 0; t < time_ms; t++)
	{
		_delay_ms(1);
	}
}
```

Appendix C - Transceiver Prototype



AviTrans

2006 ELEC499 Design Project

Digital Avalanche Rescue Transceiver

Team Members:
Kyle Weston/Mike Blarowski

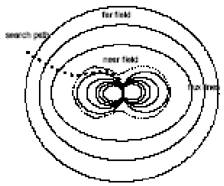
Supervisor:
Dr. Jens Bornemann

Why use a transceiver?

Travel and recreation in avalanche terrain has many inherent risks due to the fact that most of the time it is done in a remote location , far away from any emergence response and rescue team. In the event of an avalanche and resulting burial, the only chance of survival is through "companion rescue," by a member of your own party. In an event of a full burial, when no visual signs exist, **transceivers are the only proven effective way to locate a completely buried victim while he or she is still alive.**

Principles of Operation

Beacons manufactured today all operate on **457kHz.** At this low frequency, the beacons generate magnetic flux lines. This mode of operation is called the near field.



QUICK READ

1. Transceivers emit signals that can be detected by other transceivers, allowing backcountry travelers to locate companions if they become buried in an avalanche.

2. Each person in a backcountry party must wear a transceiver and turn it on to "send" mode when they set out.

3. If a party member becomes buried, the others turn their transceivers to "receive" mode to pick up the buried signal.

4. Analog transceivers can pick up a signal from a greater distance, but they require user practice to yield reliable pinpoint accuracy.

5. Digital beacons have a smaller range in which they will pick up a signal, but once they get one, they are generally faster to do a pinpoint search.

Searching Techniques

The main difference among transceivers is the search method you use to locate the target. Analog beacons with one antenna require a specific search pattern called **the grid or bracket method.** Those with dual antennas use the **induction line method,** which follows the pattern of electromagnetic energy transmitted by the buried transceiver.

Analog vs. Digital Transceivers

Analog beacons respond to signal by emitting audible beeps that grow louder as you get closer to the buried beacon. Some models, employ a visual indicator. It is often difficult to detect changes in volume when there is high wind or multiple burials.

Digital beacons use a microprocessor to translate the electronic signals into a beeping tone and a visual display. The display is either an LCD or LED panel that shows distance and/or the direction to the transmitting beacon. The audible signal of digital beacon becomes higher pitch or gets faster the closer you get to your target, whereas signals on analog models get louder. Digital beacons light up or show on the display panel when they are aligned with the magnetic flux lines transmitted by the buried beacon.

One vs. Two Antennas

One Antenna — Most of the transceivers currently on the market have one antenna. Using them requires that you keep them in a constant orientation once you pick up your signal, or your reading will not be accurate. This means that if you turn your body around, the beacon's direction must not change. It takes a little practice to get the hang of it! They also require a particular search pattern called the grid method.

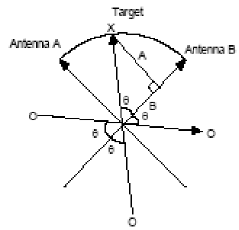
Two Antennas — Beacons with 2 antennas show when you need to adjust your position to line up with the magnetic flux lines of the transmitting beacon as you search. For one particular beacon, the display arrows light up to point you in the right direction along the flux lines. When the middle arrow lights up, both antennas are receiving equal signal strength, and you are along the flux line or the path of the strongest signal. Side arrows light up to point you in the correct direction when one antenna is getting a greater signal than the other.

AviTrans – Tracker 100

Dual antenna advantage

The Tracker100 incorporates two antennas to point you in the direction of the transmitting beacon.

Since the two receive antennas are orthogonal the direction can be calculated using the simple trigonometric relationship where A and B represent the received signal strength from antenna A and antenna B respectively. However, as can be seen from Figure 1 below, for a given angle there exist four possibilities for the true direction to the target. **We are currently working to resolve this ambiguity, possibly by adding a third antenna.**



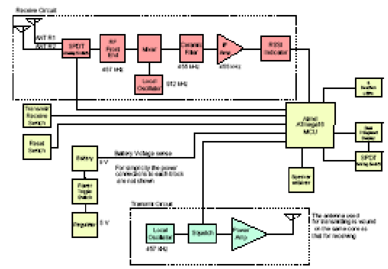
$$\theta = \arctan\left(\frac{A}{B}\right)$$

$$\text{distance} = \sqrt{A^2 + B^2}$$

Figure 1 - Direction analysis diagram. X represents the true target location and O represents false target locations.

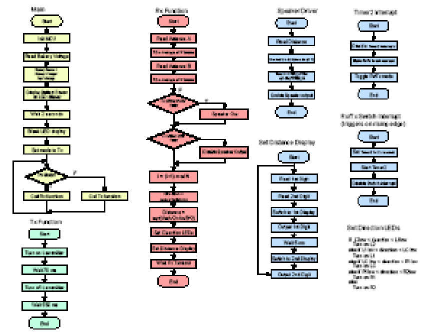
Hardware

The following Block diagram illustrates is a system level snapshot of the transceiver module.



Firmware Flow Chart

The following diagram illustrates the firmware program flow.



Cost Analysis

One of the main motivating factors behind this project was cost. Currently available avalanche radio transceivers retail for anywhere from $230 to $450. It is apparent from the table to the right, however, that an avalanche transceiver can be made for only a small fraction of that cost.

Part	Cost
MCU	$8
Oscillators	$15
PCB	$10
Enclosure	$50
Ferrite	$5
Misc. Components	$10
Total	$98

Specifications

- Carrier frequency: 457 kHz
- Receiver sensitivity: -50 dBm
- Transmit power: 0 dBm
- Microcontroller: Atmel ATmega16
- Power requirements: 30mA (180mW on a 9V battery)
- Maximum range: 35 ft