Design and Construction of a Rotating Object Detection System

MAE 4733 Semester Project

Alexandre Baillargeon

December 9th, 2011

# Table of Contents

[1 Table of Contents 2](#_Toc311446854)

[2 Introduction 4](#_Toc311446855)

[2.1 Goals 4](#_Toc311446856)

[2.2 Design Constraints 4](#_Toc311446857)

[2.3 Design Decisions 5](#_Toc311446858)

[3 Physical Phenomena 6](#_Toc311446859)

[3.1 Speed of Sound 6](#_Toc311446860)

[3.2 Range Finding Equation 6](#_Toc311446861)

[4 Hardware Design 7](#_Toc311446862)

[4.1 Overview 7](#_Toc311446863)

[4.2 PIC Microcontroller Board 7](#_Toc311446864)

[4.3 Signal Generation 8](#_Toc311446865)

[4.4 Transmission 8](#_Toc311446866)

[4.5 Signal Reception/ Amplification 9](#_Toc311446867)

[4.6 Comparison/ Tone Decoding 10](#_Toc311446868)

[4.7 Motor Control 11](#_Toc311446869)

[5 Software Design 13](#_Toc311446870)

[5.1 Initialization 13](#_Toc311446871)

[5.2 Timer0 ISR 13](#_Toc311446872)

[5.3 External ISR 13](#_Toc311446873)

[5.4 USART 14](#_Toc311446874)

[5.5 MATLAB Code 14](#_Toc311446875)

[6 Results 15](#_Toc311446876)

[6.1 Object Layout 15](#_Toc311446877)

[7 Further Development 16](#_Toc311446878)

[7.1 Beam Focusing 16](#_Toc311446879)

[7.2 Software Improvements 16](#_Toc311446880)

[7.3 Hardware Fine-Tuning 16](#_Toc311446881)

[8 Applications 17](#_Toc311446882)

[8.1 Automated Map Generator 17](#_Toc311446883)

[8.2 Robotic Lawn Mower 17](#_Toc311446884)

[9 Acknowledgements 18](#_Toc311446885)

[10 References 19](#_Toc311446886)

[11 Code Listing 20](#_Toc311446887)

**Table of Figures**

[Figure 1 - Rotating Radar on Military Ship 4](#_Toc311446888)

[Figure 2 - Distance from Echo 6](#_Toc311446889)

[Figure 3 - PIC Board Construction 7](#_Toc311446890)

[Figure 4 - Astable 555 Timer Diagram 8](#_Toc311446891)

[Figure 5 - Transmitter Circuit Diagram 9](#_Toc311446892)

[Figure 6 - Transmitter/Receiver Unit 9](#_Toc311446893)

[Figure 7 - Comparator Output 10](#_Toc311446894)

[Figure 8 - Tone Decoder Circuit 11](#_Toc311446895)

[Figure 9 - Tone Decoder Output 11](#_Toc311446896)

[Figure 10 - Stepper Motor High-level Diagram 12](#_Toc311446897)

[Figure 11 - Object Layout 15](#_Toc311446898)

[Figure 12 - Transducer Beam Angle 16](#_Toc311446899)

[Figure 13 - Microsoft Kinect Peripheral 17](#_Toc311446900)

# Introduction

## Goals

In today’s world, object detection finds a use in many different technologies ranging from plane detection on a large scale such as airports, to weather and cloud detection for predictions. One extremely well-known implementation of an object detection technology is radar. The physics that govern radar object detection are well-known and understood, but the general idea is to use specialized electromagnetic waves (radio or microwaves in the case of radar) to bounce waves off objects. The purpose is usually to detect objects around a particular center. In this project, the goal was to design and construct a radar-like detection device: namely, a device that would be able to detect objects relatively close to itself by performing a “scan” of its surroundings. Due to its low-cost nature, the accuracy of the planned system was unknown, but a major goal of the project was for the operator to be able to recognize objects when placed around the system. The system is to be controlled by a microcontroller in order to demonstrate some of the abilities and techniques learned in class.



Figure 1 - Rotating Radar on Military Ship

## Design Constraints

In order to limit the complexity to be able to construct an actual proof-of-concept in the required time (a little over one month), the design had to be simple. While radar has been proven as a mature technology, the complexities in dealing with radio or microwaves ruled out any such designs. Additionally, research revealed that very accurate systems have been developed by using electromagnetic waves in the visible or near-visible spectrum in range finding applications. However, the high performance electronic systems needed to measure the time taken for these electromagnetic waves to return was daunting from both a financial and electronic design perspective.

Thus, the decision to use sound waves was chosen. Due to the low speed of acoustic waves in air, determining transit times for pulses became much simpler with low cost hardware available to a student. However, acoustic waves are prone to reflection and absorption, which may limit their usefulness for range finding certain materials.

## Design Decisions

As stated in the previous section, the final design will use acoustic waves in order to calculate the distances to objects around it. Specifically, the design will use ultrasonic sound waves due to their attenuation characteristics. As frequency of a wave in air increases, the higher attenuation provided by the air provides less background noise at that particular frequency. Ultrasonic waves are defined as sound waves above the threshold of human hearing, which is approximately 20 KHz. In the case of this design, the center frequency chosen for the waves will be 40 KHz, due to the availability of low-cost transducers at that particular frequency. Controlling the rotation of the platform will be accomplished by the use of a stepper motor due to its inherent non-slip nature and simple angle calculations.

A microcontroller will control a pulse generator in to generate and transmit a short “burst” of acoustic energy, and will be able to measure the amount of time that elapses before the return echo is heard. As acoustic velocity within a particular medium is relatively stable without temperature changes, the distance to the object which causes the echo can be determined. These physical phenomena will be discussed in a later section.

To compartmentalize the development of the object detection system, the design and construction will be broken up into the following steps:

* Transmitter/Receiver Circuit Design
* Microcontroller Integration
* Embedded Range Finding Software
* Rotating Platform Design
* PC Plotting Software

# Physical Phenomena

## Speed of Sound

The speed of sound in air is crucial to the accuracy of the design. For ideal gases, the speed of sound equation is where *k* is the heat capacity ratio, *R* is the molar gas constant, and *T* is the temperature in Kelvin. For air at room temperature, the nominal value of *c* is approximately *343 m/s*.

## Range Finding Equation

Once the speed of sound in a medium is known, a simple relationship between speed and time can be used: where *Vbar* is the average velocity and *t* is the time elapsed. Therefore, in the case for air a value of *343 m/s* can be used in case of *v*. However, it is important to consider that because the acoustic waves must travel to and from the object that produces the echo, the distance calculated is the *total travel distance*. The distance from the transmitter to the object, then, is half of this total distance.

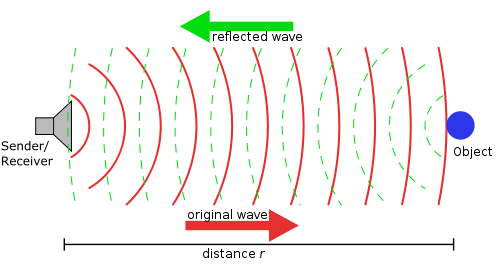


Figure 2 - Distance from Echo

# Hardware Design

## Overview

To accomplish the task of calculating distance by measuring the transit time of acoustic waves, a circuit must be designed to generate sound and listen for its return. Eliminating false triggers such as background noise must also be considered. In order to maximize the transmission power and receiving sensitivity, the nominal positive rail was chosen to be +12V.

The hardware design and construction of this system will be split up into multiple sections in order to ease in planning and troubleshooting, respectively. The sections will be discussed as follows:

* PIC Board
* Signal Generation
* Transmission
* Signal Reception/ Amplification
* Comparison/ Tone Decoding
* Motor Control

## PIC Microcontroller Board

A PIC microcontroller board was designed and constructed for educational purposes. Microchip had responded to a sample request for microcontrollers (2 18F4520s and 2 18F4550s), and provided the 40 pin DIP chips. The result was a much simpler board than anticipated: a protoboard which contained a +5V regulator, an 8 MHz crystal oscillator, and a PIC 18F4520 on a 40-pin socket. In order to program the PIC while in its socket, a 5 pin in-circuit-serial-programming (ICSP) header was added. Later, a MAX232A salvaged from an old motherboard was soldered onto the USART TX/RX pins of the PIC.

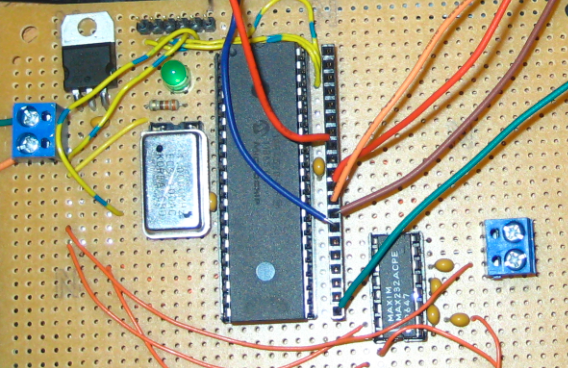


Figure 3 - PIC Board Construction

## Signal Generation

The first step in measuring transit time of a wave is the generation of the wave itself. In the design specifications, a 40 KHz signal is required. A 555-timer was used for the signal generation due to its simplicity and extremely low cost. Figure x shows the 555 timer circuit used.

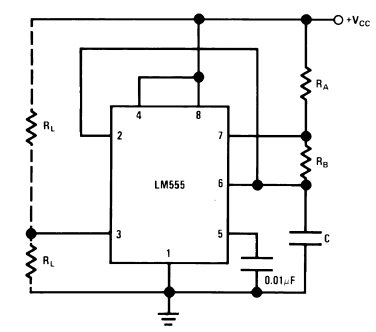


Figure 4 - Astable 555 Timer Diagram

A 555 timer connected in this mode is said to be in *“astable operation.”* As the capacitor on pin 6 charges and discharges, it produces a square wave of adjustable frequency and duty cycle on the output (pin 3). The equation provided in order to determine the frequency of the 555 timer in astable operation is . By using a finely adjustable cermet potentiometer in series with a resistor for Rb, the frequency of the timer circuit can be fine-tuned by adjusting the potentiometer and checking the output with an oscilloscope. In the actual circuit, the reset pin (pin 4) is connected to the microcontroller instead of Vcc. The pin is kept low until a pulse needs to be sent, when it is briefly (~ 1ms) brought high. This only allows the 555 timer circuit to generate the 40 KHz signal for the duration the reset pin is kept high.

## Transmission

The next step in measuring transit time is the actual transmission of the wave. Once the pulse is generated by the timer circuit, the output pin of the 555 is connected to a 2N2222 NPN transistor via a 1K current limiting resistor, here represented as R2.

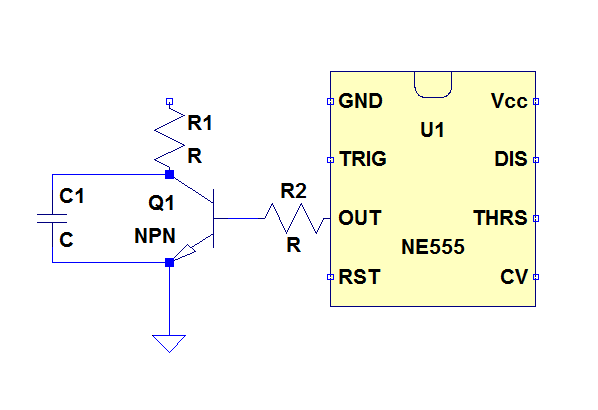


Figure 5 - Transmitter Circuit Diagram

The ultrasonic transducer (*Jameco ValuePro 40T-12B*) is shown as an equivalent capacitance C1, and is energized as the 555 timer output pin oscillates at 40 KHz. In essence, the transistor functions as a switch which controls the +12V provided by the power supply being applied to the transmitter.

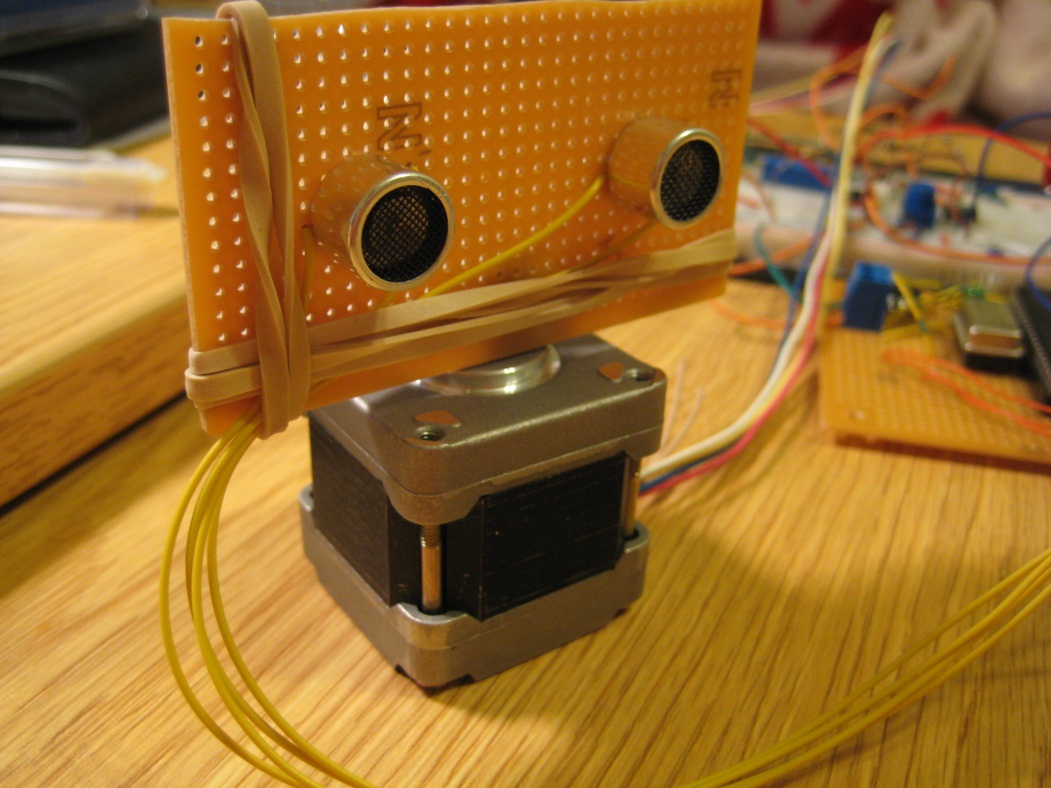


Figure 6 - Transmitter/Receiver Unit

## Signal Reception/ Amplification

Once the signal is transmitted by the ultrasonic transducer, the next step is to receive the bounced signal. This process is first done by using the Jameco ValuePro 40R-12B ultrasonic receiver. The receiver also has its center frequency at 40 KHz, making it well suited to picking up signals from the matching transducer.

This receiver is connected to a high-speed operational amplifier in order to amplify the received signal. By trial and error, the gain in order to “see” a signal without clipping was found to be approximately 500. One important design consideration is that due to the nature of a non-ideal operational amplifier, the initial choice of using a LM741 op-amp did not produce acceptable results. The gain-bandwidth product (GBWP) of the LM741 is 1 MHz, and distorted the signal at the frequency and gain required for the design. Therefore the LT1022 op-amp was chosen as it performed better at the high speeds/gain required. In order to eliminate the dual rail power supply requirements of the op-amp, the ultrasonic transducer was biased to a +6V virtual ground.

## Comparison/ Tone Decoding

At this stage of the circuit, a suitably large signal is present (approx. 2V peak-to-peak) on the output of the previous stage. The purpose of the Comparison/ Tone Decoding stage is to determine whether or not the signal seen on the input is the signal that was generated earlier.

Using a comparator allows the setting of a “threshold” that must be crossed in order to send the output high. In figure x, there are two signals of note: the high amplitude pulse where the receiver directly “hears” the transmitter due to their physical proximity; and the echo returning at the end of the trace. Note that the bottom signal is not DC, the solid color indicates it oscillating from logic LOW to HIGH at a frequency to be determined.

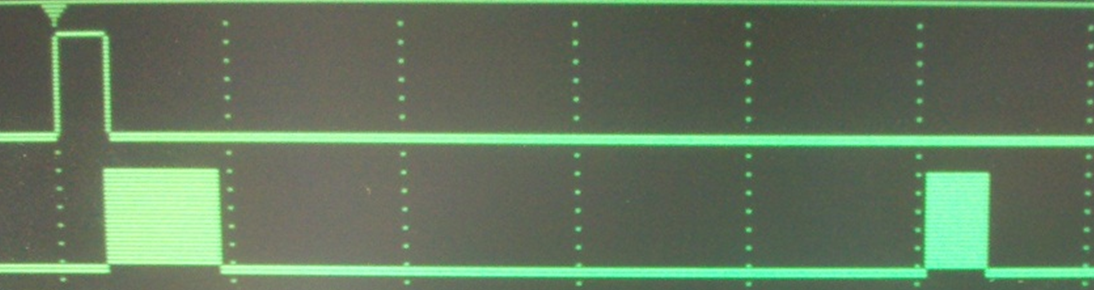


Figure 7 - Comparator Output

The final stage of the receiving circuit is to determine the frequency of the signal. In order to eliminate false triggers by other noise sources, a “tone decoder” IC is used. In this case the LM567 was chosen due to its very low cost and ease of use. Figure x shows the circuit used: it was labeled as a 100 KHz test circuit and taken from the application notes section of the LM567 data sheet. Slight modifications were made to R1 and C1 in order to change the center frequency to 40 KHz.

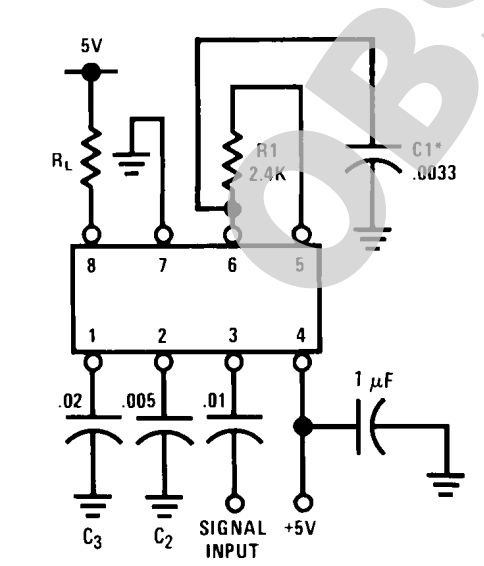


Figure 8 - Tone Decoder Circuit

The LM567 requires a +5V input supply, and therefore its own voltage regulator. In this case, the signal produced from the comparator enters the input (pin 3) and keeps the output (pin 8) high until the correct signal is detected. If the correct frequency is found, it pulls the output (pin 8) low. The clean signal will allow the use of a falling edge interrupt to determine the exact time the echo returns. Figure x shows the completed circuit functioning: D0 is the input trigger; D1 the comparator output; and D2 the tone decoder output.

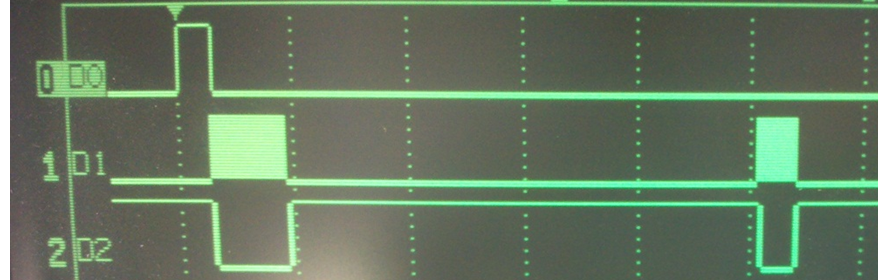


Figure 9 - Tone Decoder Output

## Motor Control

The last hardware component needed to build the object detection system is the motor controller. Due to cost limitations, the stepper motor used was salvaged from an obsolete scanner. The motor had 6 leads, and was therefore a unipolar stepper design. However, by ignoring the center taps it can be used as a bipolar stepper motor giving it more torque. Markings on the side of the motor indicate a step size of 1.8 degrees, giving the motor 200 steps/revolution. A L293NE H-bridge IC controlled the high currents to drive the motor.

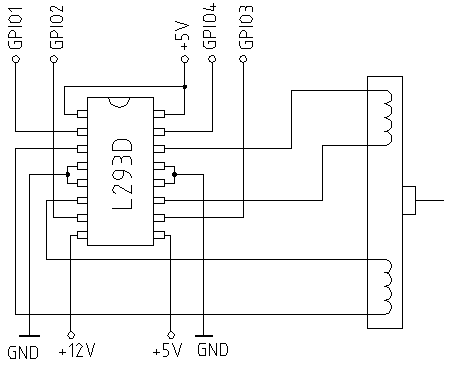


Figure 10 - Stepper Motor High-level Diagram

# Software Design

The software running on the microcontroller was programmed in MPLAB. MPLAB and the PIC 18 series C compiler, C18, were available for free to those in an academic setting. Using the MPLAB compiler brought with it new complexities that had not been considered in class, but also new understanding with regards to the abstraction provided by the CCS C compiler.

The general flow of the program is as follows:

1. The configuration registers are set
2. Output to pulse generator is set high
3. Timer0 ISR is enabled
4. Short wait to widen output pulse
5. Output to pulse generator is set low
6. Long wait to prevent false trigger (caused by receiver picking up transmitted signal)
7. External ISR is enabled
8. Infinite wait until external ISR is triggered
9. External ISR stops timer
10. Print timer0 to USART
11. Move motor to the next step
12. Return to #2

The major software design decisions will be discussed in their respective sections.

## Initialization

Initialization includes the start-up sequence of events that occurs just after the microcontroller is released from reset. In this case the sequence consisted of enabling the oscillator, setting the input/output tri-state registers (TRIS), and configuring the interrupt service routines (ISR).

## Timer0 ISR

Timer interrupts were crucial for the measuring of the transit time of the wave. One of the important ideas for the microcontroller software was the correct selection of the timer0 interrupt. Instead of choosing an arbitrary timer (timer overflow every 1ms, 0.1ms, etc), the decision was made to overflow using the time for a wave to travel 1cm in air. This simplified distance calculations, as the timer0 value became the distance the wave traveled. This necessitated the use of an 8 bit timer, and the frequency was calculated by using

Solving for x, an 8-bit timer0 interrupt with an initial value of **23** at 32 MHz clock speed will overflow every 1 cm traveled in air.

## External ISR

The use of both timer and external ISRs brought with it the concept of priority interrupts. The INT0 external interrupt is always a high priority ISR, but the timer ISR can be changed using its configuration register (T0CON). Enabling dual-priority interrupts allowed the timer to run until the external ISR stopped it. Setting *INTCON2.INTEDG0* to 0 allows the interrupt to trigger on the falling edge produced by the tone decoder. An important consideration is that each interrupt has a flag bit used to determine which interrupt caused the jump to the ISR, which *must be reset after every interrupt*. Failure to do so will not allow the interrupt to be triggered the next time, and will call the interrupt immediately if it is disabled and enabled.

## USART

The PIC 18F4520 has a dedicated internal serial communication module known as the Universal Synchronous Asynchronous Receiver Transmitter (USART). The USART is used to communicate with serial devices, and when paired with a line driver can be used to output RS-232 to a computer or other peripheral.

One of the complexities involved with using the USART for RS-232 is the generation of a baud rate. There are no baud rates to choose from in the control registers: a dedicated timer is provided and must be configured to provide the right baud rate. In this case, a baud rate of 57.6 Kbps was selected:

where Fosc is the clock speed (32 MHz). Once the USART was configured with n=8, any printf statements made were directed to the computer connected by serial cable.

## MATLAB Code

Since one of the primary goals of the project was to be able to plot the results to show the objects surrounding the system, a MATLAB script was developed. In order to keep the script simple, serial communication input was not included in the MATLAB code; any serial communication from the circuit was copied and pasted from a Putty serial terminal into a CSV. These values were read in using MATLAB’s *fileread* and *textscan* functions.

# Results

Results from each stage of the design process were verified independently before combining them into a plot. The distance ranging circuit was tested by placing it approximately a distance of 1 meter facing a wall. Similarly, the stepper motor control was tested independently by rotating the motor between step 0 and step 199. Once the individual systems were verified, they were assembled together and the object plotting could be tested. Due to the way the data is transmitted (continuously and stored as a CSV), the system can be enabled and will acquire data for any length of time, provided the system does not move.

## Object Layout

Testing for ”hard” objects was the first step. Hard objects encompasses objects with a reasonably specular surface such as books, cans, boxes, etc. These are to be contrasted with “soft” objects such as clothing, drapes, curtains, foam, and other objects expected to absorb sound waves.

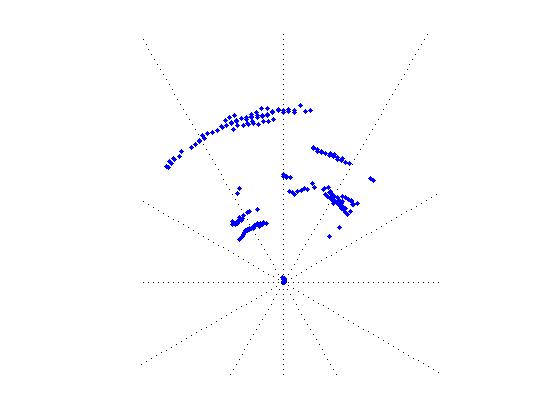


Figure 11 - Object Layout

Figure x shows the results from a 90 degree sweep of a box and canister. In the upper left region a chair sits beyond the table, and may be responsible for the flat reflection present on the scanned image. This image, while very rough, does indeed meet the original goal of being able to plot the results onto a graphical interface.

Further testing on different geometries would be beneficial.

# Further Development

## Beam Focusing

One of the problems with using the specific acoustic transducers to determine distance was the focusing of the acoustic energy. Ideally, the wave would be a point into space traveling towards the object to “see”, and a small portion of that energy would be reflected towards the receiver. However, the reality is that the waves are transmitted as cones. While the inner cone has the strongest signal, much of the energy is spread out into a large cone. By developing a “cone“ or “horn”, the signal could be focused into a much smaller cone and decrease the likelihood of false echos.

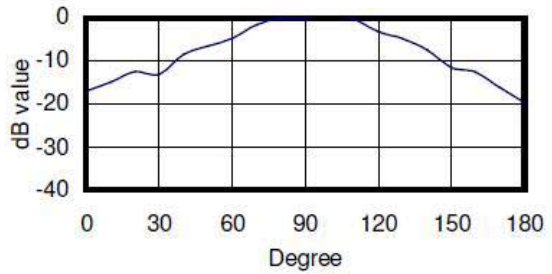


Figure 12 - Transducer Beam Angle

## Software Improvements

Significant improvements could be made in software to simplify the plotting of the points on a computer. Instead of taking one measurement and moving to the next step, multiple measurements would be taken and an algorithm to average would be considered. An algorithm is mentioned because in some cases the readings might be as follows (in cm): 25,25,24,3200 (timeout), 25. In this case the 3200 would be thrown out because the average of these values would not be accurate.

## Hardware Fine-Tuning

Although the hardware functioned correctly, certain changes might be made to increase the performance without sacrificing cost or accuracy. Different load resistors would be put in series with the transmitter to attempt to increase the power transmitted and therefore received. Furthermore, an additional amplification stage to increase in the gain to 1000 would be helpful. The problem here is that the gain was designed to avoid clipping of the initial signal, not the echo.

# Applications

Object detection systems have applications in almost every industry. Long range systems can detect ships and planes to avoid collisions, while short range systems can detect smaller objects such as trees or humans. The focus on applications for the type of system developed in this project will focus on short ranges.

## Automated Map Generator

The pinnacle of object detection technology is to build a system accurate enough to scan surfaces to a reasonable degree of accuracy while handling any stray echos or beams. In this application, a robot would be fitted with the object detection system in addition to wheel encoders and a compass. The robot would use the object detection system in addition to its encoders to measure distances as it travels around a room or building. Given enough time and memory, it would have a complete procedurally generated map. A very interesting project with exciting algorithms needed to develop relationships between the distance inputs and the wheel movements.

Instead of using ultrasonic frequencies for this application, a laser would work better due to its lack of reflection patterns around corners. Alternatively, research has shown a Microsoft product, the Kinect, to be able to sense distances by using infrared projection technology. Open source development is very active for the Kinnect.

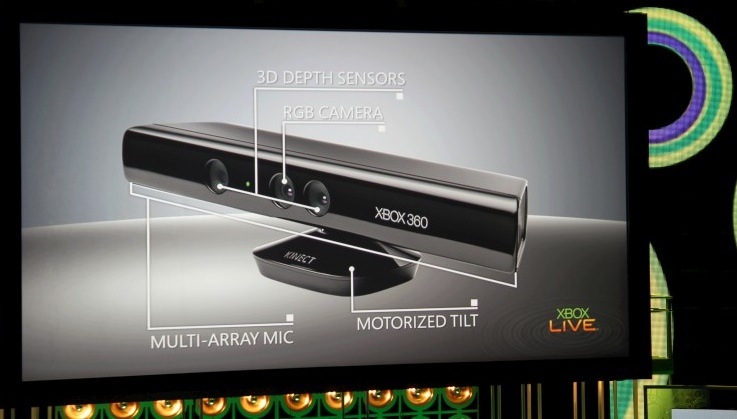


Figure 13 - Microsoft Kinect Peripheral

## Robotic Lawn Mower

Another application for a rotating object detection system is in a robotic lawn mower. The system could be mounted at the front of the robot and when encountering a tree or other object, could determine the best path to go around the object. By rotating itself and looking backwards, it could make sure it had avoided the object.

# Acknowledgements

In the design and construction of this project I received assistance from many sources. I’d like to thank

* Linear Technology for providing sample LT1011 comparators and LT1022 op-amps
* Microchip for providing the PIC 18F4520 used in the project
* Michel Baillargeon and Siemens AG for providing the Pickit 2 Programmer and Oscilloscope for troubleshooting
* Dr. Young and Chris for helping in the mechatronics lab

# References

1. Jameco, “Valuepro Ultrasonic Transducer Datasheet” <http://www.jameco.com/Jameco/Products/ProdDS/139492.pdf>
2. Linear Technology, “LT1022 Operational Amplifier Datasheet” <http://cds.linear.com/docs/Datasheet/lt1022fa.pdf>
3. Microchip Technology, Inc, “PIC 18F4520 Datasheet” <http://ww1.microchip.com/downloads/en/devicedoc/39631a.pdf>
4. Moran & Shapiro “Fundamentals of Engineering Thermodynamics, 6th Edition”
5. National Semiconductor, “LM567 Datasheet” <http://www.national.com/ds/LM/LM567.pdf>
6. National Semiconductor, “LM555 Timer Datasheet” <http://www.national.com/ds/LM/LM555.pdf>
7. Romero, Dana, “Radar” < http://scienceworld.wolfram.com/physics/Radar.html>

# Code Listing