Project Proposal

Realtime Indoor Positioning System

The Bit-Shifters

Jordan Squire
Colby Litnak
Jake Willoughby
Paul Willoughby

http://groups.google.com/group/bit-shifters/web

Table of Contents

Introduction & Motivation

The proposed project is a three-dimensional object tracking system. This system will be able to identify the accurate 3D location of a tagged object in a room. Knowing an accurate location of an object has many possible applications. The intended demonstration for this project is a system where a person can wear a transmitter on their head and they will be facing a 3D scene presented on a computer monitor. As the person moves toward the monitor, the image on the screen would adapt in such a way to make the person feel like the image on the screen is a 3D scene.

This project will use an array of microphones at a base station tuned to pick up specific high frequency, inaudible sound waves coming from a transmitter on a trackable entity. The signal will pulse periodically allowing the base station to calculate its location and track the entity as it moves. Each microphone will receive the pulsed signal at a different time based on the speed of sound and the distance between the microphones. Using the differences between these times of arrival and the known locations of the microphones in the array, the base station will be able to accurately calculate the location in three-dimensional space of the transmitting entity. This calculation is called multilateration. Because sound bounces off of walls, the bounced signals will have to be ignored, introducing the constraint on the system that the transmitter must be within line of sight to guarantee accuracy. The base station will include a software program that performs the mentioned calculations and displays the 3D coordinates of the object to a screen at a minimum. If time permits, additional software such as a 3D scene as explained above or multilateration graphs will be added to the software system.

This project is motivated by the desire to learn the technology involved and participate in the creation of a practical system. This system will use high frequency sound transducers, microphones, amplifiers, filters, and a microcontroller. Team members will learn how to implement a complicated system and increase understanding of real life projects.

Components and Interfaces

Transmitter

The transmitter system consists of a power source, a frequency generator, and an ultrasonic (high frequency) transducer. The frequency generator will be used to "pulse" power to the transducer. This signal may need to be amplified. The transducer, when receiving power, will send 40 kHz sound signals over the air. The transducer dB levels are linearly proportional to the voltage applied. The voltage level will be configured after further tests are done, but preliminary tests indicate that the transmitter does not need to produce a very loud signal. More tests will be done to investigate how to minimize the intensity of the signal to decrease any potential health risks. The transducer can produce over 100dB signals if absolutely necessary. The frequency generator will be configured to pulse at 20Hz, because of microphone placement constraints (explained below) as well as concerns that the rest of the system may not be able to keep up with any frequency faster than that. The length of the pulse will be 325 µs (about 13 sound waves). See figure 1 for a high level diagram of the transmitter system.

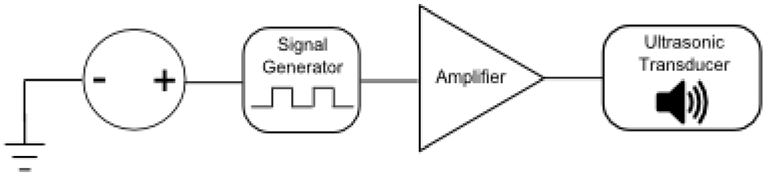


Figure 1. Block Diagram of Transmitter System

Receiver

As the transmitter sends the ultrasonic signals, the receiver system picks up these waves and triggers the input capture on a microcontroller. The receiver system has four microphones (also transducers) that convert the sound waves into an electrical signal. Each of the four signals will go through a few stages. First the signal will be amplified to about twice the signal. The second stage is a log amplifier which makes the output voltage a multiple of the natural log of the input voltage. The log amplifier is an attempt to amplify weaker signals without amplifying larger signals too much. The signal is then passed through an integrator circuit which will smooth out the 13 sound waves into one larger curve. The signal then passes through another amplifier so that it triggers a monostable multivibrator. The multivibrator "triggers" when the input voltage reaches a certain level and amplifies it to a rail voltage of 5 volts. This signal will be held for 40ms, which is long enough for bouncing sound signals from the same source to the transmitter to dissipate. The circuit will be ready for another signal approximately 10ms before the next transmitter pulse will be received. The purpose of this also serves to decrease the possibility of ghost signals. The multivibrator has Schmitt trigger inputs to effectively debounce the signal. LEDs will be added at this stage to potentially help with debugging. The four signals at this point will be 5V digital signals compatible with the microcontroller.

A 20Hz pulse rate is slow enough that microphone placement will have loose constraints. Specifically, a signal traveling the speed of sound (343 m/s) must reach each microphone within 50ms of each other or within 17 meters of each other, in order for the software to tell which timer values are associated to the signals processed. The configuration and placement of the microphones can change, but the configuration currently selected is to have one microphone at a relative "origin" and one on each of the three axis’s at different distances from the origin (see figure below). This configuration removes the possibility of multiple points in space having the same distance from all microphones, as well as making multilateration calculations easier. Microphone placements that have any two microphones equidistant from another introduce the problem indicated above. Another extremely loose constraint on microphone placement is explained in detail below.

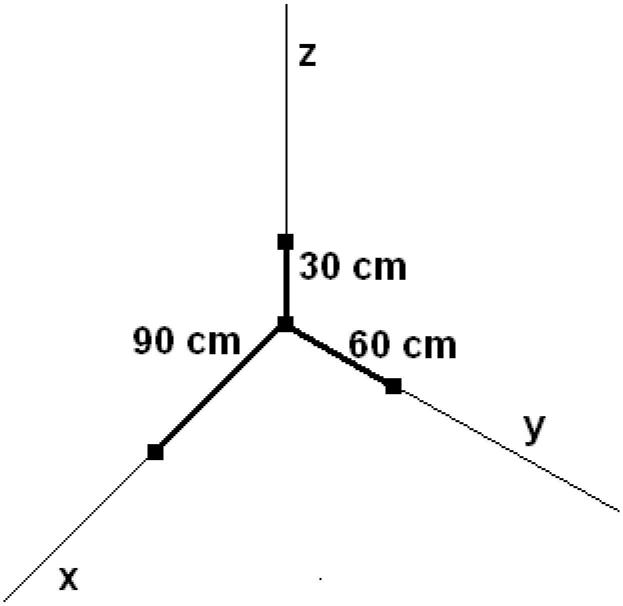


Figure 2. Microphone Placement Configuration

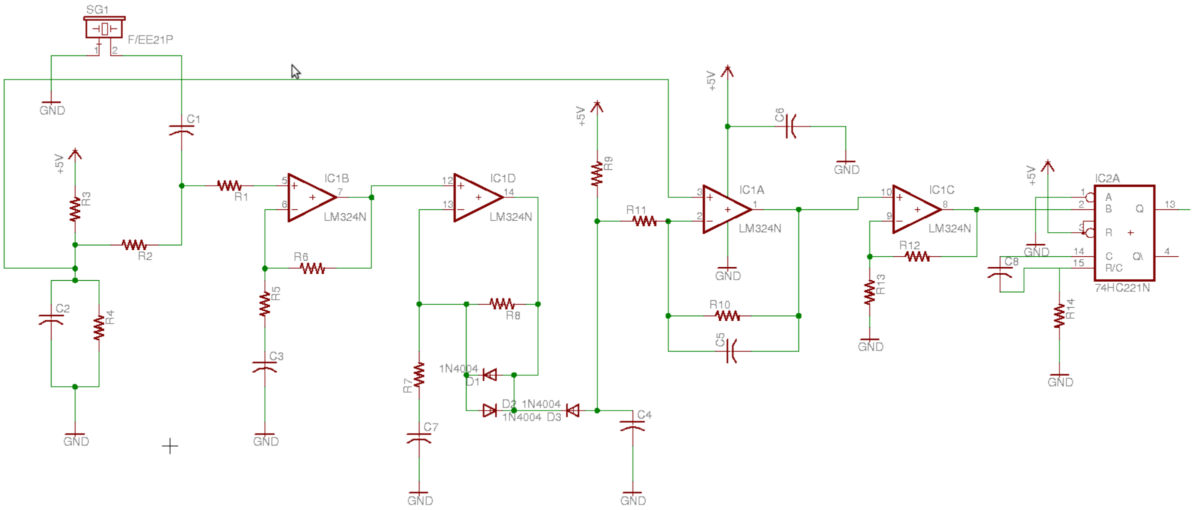


Figure 3. Receiver System Schematic

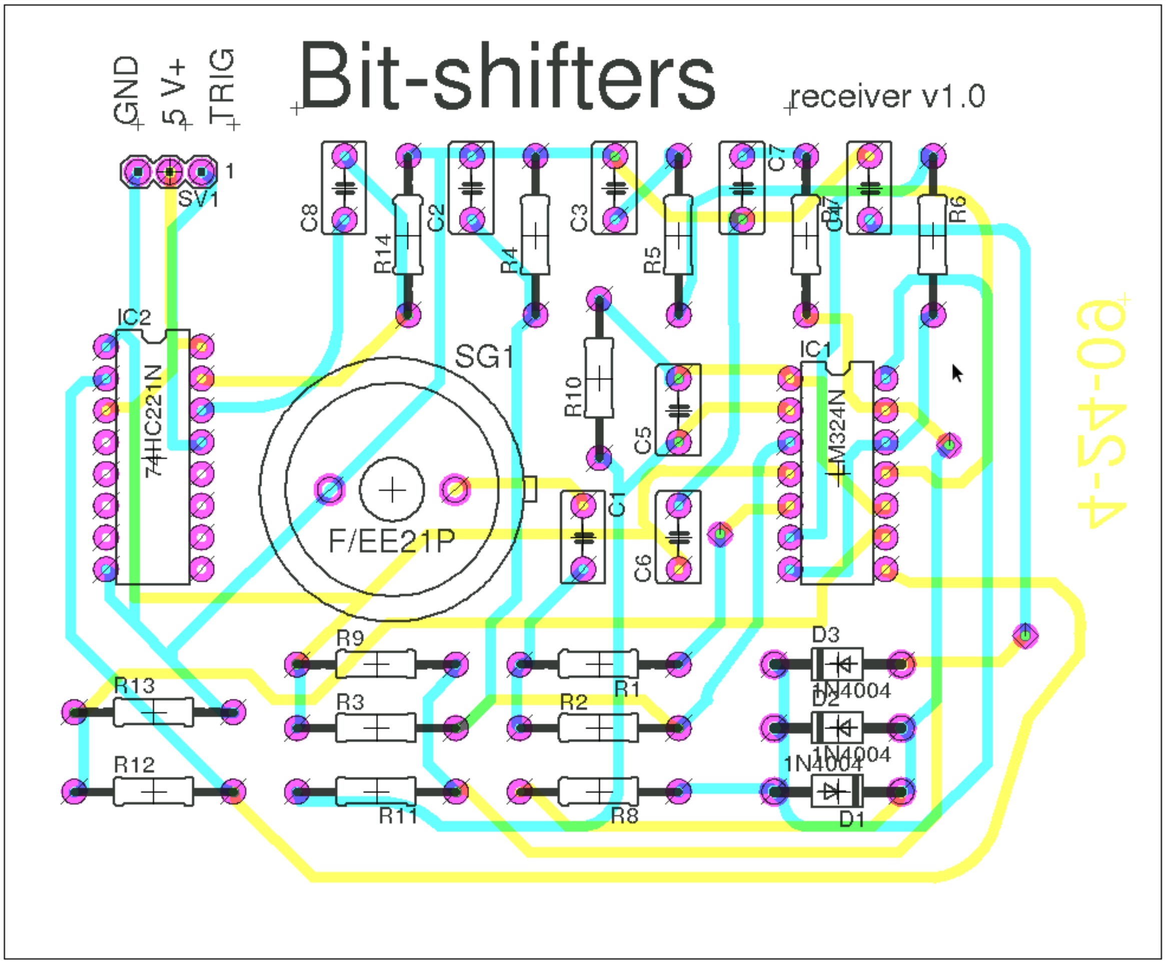


Figure 4. Receiver PCB Layout

Microcontroller

The microcontroller system acts as a data acquisition system, which acquires the time differences from the incoming signals, latches internal timer values, and sends the data to a computer attached via USB. The microcontroller "listens" for pulses via four input capture pins configured to latch the values of a 32-bit on board timer. These pins will also have pull-down resisters, so that when the 5V signal from the multivibrator is received, built-in hardware will latch the value of the timer into a queue. The main loop of the microcontroller will simply check to see if each of the four values have latched a timer value, and then send the four values to the computer. The microcontroller board has an on board USB interface that makes the device appear as a COM device to the computer. Once the four values are received by the computer, the time differences are calculated and delivered to the software / demo portion of the project.

The 32-bit timer will be configured to run at 80 MHz. In order for the timer values to mean anything, they must not turn over more than once in a given pulse period. This means that 2,147,483,648 clock ticks at a maximum can pass between the longest possible distance between any two microphones. This represents the worst case. Based on the speed that sound travels, the largest spacing our receivers can be placed is 15.1 Miles apart, as it takes approximately 26 seconds for timers to count to about 2 million.

The microcontroller will handle ghost signals by emptying out each of the input capture queues after a set of four timer values are sent to the computer. Additionally, if a missed signal occurs, the microcontroller will effectively ignore any incomplete set of timer values. If all four input captures have not fired from first latch after 40ms, the microcontroller empties out the input capture queues.

The board used will be the UBW32 board which is an inexpensive development board for the Microchip PIC32 that handles the USB interface. The board contains a PIC32MX460F512L microcontroller, and a female USB mini-B connector pre-connected to the appropriate pins on the Microcontroller. The board allows for DIP style bread boarding. As mentioned above, firmware will be used that makes the board appear as a COM device on the computer. The PIC32MX460F512L microcontroller has an 80 MHz clock along with five 16-bit timers, with a configuration of chaining two of them together as a 32-bit timer. The chip also has five input capture pins, only four of which will be used. These pins are directly available from the UBW32 board. It has 512K of flash and 32K of RAM. Initial analysis of firmware size plus customization that needs to be done indicates that this is plenty of memory for what needs to be done. Default USB firmware takes up 2K, and customizations will be 1K at most. The CPU is has a RISC MIPS architecture. Each Input Capture has a 5.5V max (3.3V+ is considered to be a high signal). The receiver system will deliver a compatible digital signal with low as 0V and high as 5V. See Table I indicating the pins on the microcontroller and how they are broken out on the UBW32 board and figure 5 for the pin out of the PIC32 microcontroller to reference the table.

TABLE I. INPUT CAPTURE AND USB PINS ON MCU AND UBW32

Name	Symbol	MCU Pin	UBW32 Pin
Input Capture 1	IC1	68	D8
Input Capture 2	IC2	69	D9
Input Capture 3	IC3	70	D10
Input Capture 4	IC4	71	D11
USB Diff +	D+	57	USB Mini-B Connector
USB Diff -	D-	58	
USB Power	VUSB	55	
USB Bus	VBUS	54	

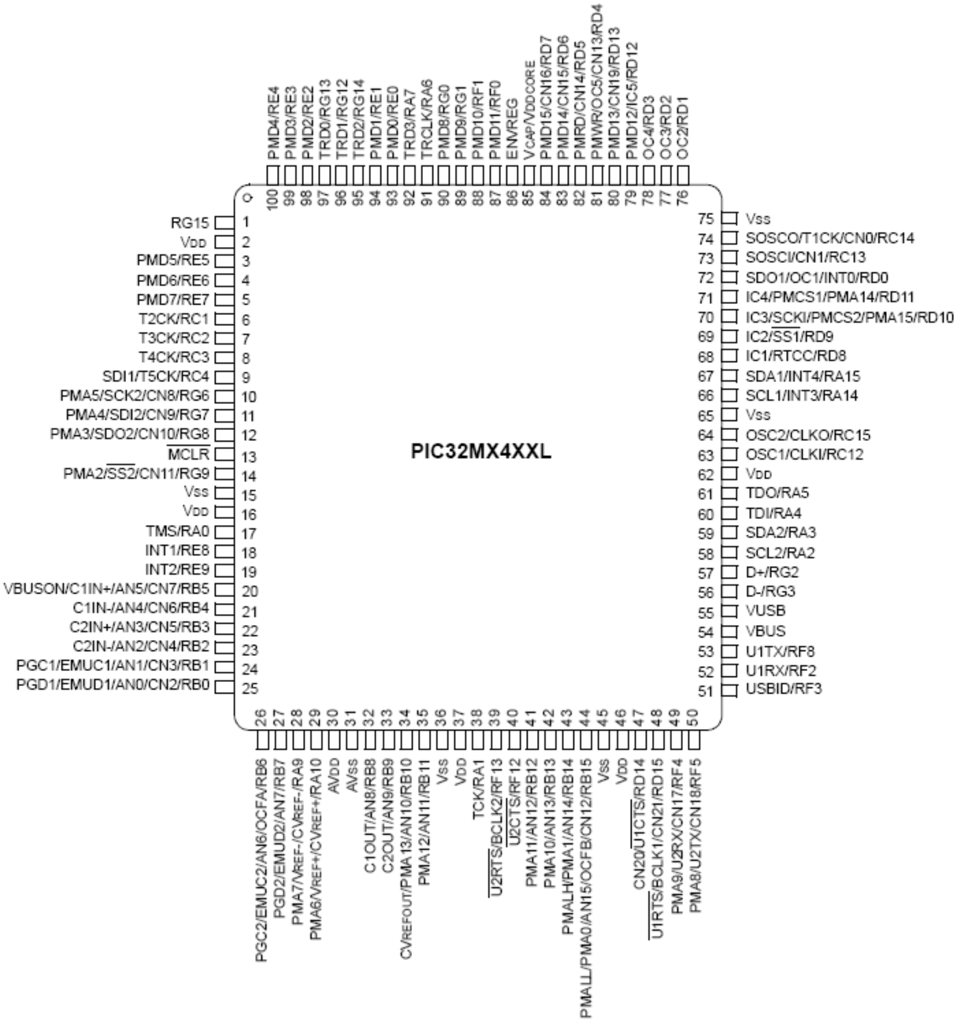


Figure 5. PIC32MX460F512L Pin out.

Multilateration

As mentioned above, the computer will see the entire system as an RS232 serial device. The computer will be running a C# .NET application that reads the data in on the serial port. This application will prepare waves of timer values for the multilateration stage.

Multilateration uses time difference of arrival of a signal. The principle is that a signal will arrive at different locations at slightly different times. The system is designed to capture four of these times at different receiver locations. Each of the travel times of a signal to these locations is simply the distance formula divided by the speed of sound.

$$(1)\ T_a = \frac{1}{c}\sqrt{(x - x_a)^2 + (y - y_a)^2 + (z - z_a)^2}$$

$$(2)\ T_b = \frac{1}{c}\sqrt{(x - x_b)^2 + (y - y_b)^2 + (z - z_b)^2}$$

$$(3)\ T_c = \frac{1}{c}\sqrt{(x - x_c)^2 + (y - y_c)^2 + (z - z_c)^2}$$

$$(4)\ T_d = \frac{1}{c}\sqrt{(x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2}$$

Where the point (x,y,z) is the unknown point of the transmitter, c is the speed of sound, and (x_i, y_i, z_i) are the respective locations of receivers at known locations. To simplify the problem, the assumption is made that receiver *a* is at the origin of the system.

$$(5)\ T_a = \frac{1}{c}\sqrt{x^2 + y^2 + z^2}$$

The time differences of the remaining receivers are calculated from the origin receiver.

$$(6)\ \tau_b = T_b - T_a = \frac{1}{c}\left(\sqrt{(x - x'_b)^2 + (y - y_b)^2 + (z - z'_b)^2} - \sqrt{x^2 + y^2 + z^2}\right)$$

$$(7)\ \tau_c = T_c - T_a = \frac{1}{c}\left(\sqrt{(x - x'_c)^2 + (y - y'_c)^2 + (z - z'_c)^2} - \sqrt{x^2 + y^2 + z^2}\right)$$

$$(8)\ \tau_d = T_d - T_a = \frac{1}{c}\left(\sqrt{(x - x'_d)^2 + (y - y'_d)^2 + (z - z'_d)^2} - \sqrt{x^2 + y^2 + z^2}\right)$$

Where (x'_i, y'_i, z'_i) is the location of the receiver with respect to the origin receiver. Each equation represents a hyperboloid which theoretically intersects at a single point (see figure 6 for an example of the three intersecting hyperboloids). Equations 6, 7, and 8, together, also represent a non-linear systems of equations with three unknowns, which can be solved to a point in the case where the timer values are perfectly accurate, or a three-dimensional volume where the point must lie, which represents the case in which timers values are inaccurate. It is expected that timer values received will be at least a little inaccurate. Thus, the system will likely use the least squares method for the non-linear systems of equations to approximate the three points.

Given the timer values, the known frequency of the timer (80 MHz), the position of the receiver microphones, and the speed of sound three coordinates will be produced indicating the 3D location of the transmitter. The system will at the very least show these coordinates on a screen.

If the system functions as planned and the location of the transmitter is accurately found and displayed, the following additional functionality may be added to the system for the purpose of presenting applications of a real-time indoor positioning system.

The C# .NET application that communicates with the microcontroller and calculates the position of the transmitter will be expanded with a socket server that will broadcast the position to listening clients. This server will allow multiple socket clients running demo applications to connect to it and receive the location of the object. Any number of applications can be started and stopped without interruption of the server. Several possible client applications will be explored and, if time permits, will be implemented.

One possible client application is head tracking. Head tracking will be done by mounting the transmitter on a baseball cap or some similar headgear, which is worn by the user. Using the knowledge about the location of the user's head, the screen can be updated to simulate looking through a window into another room. This room can be filled with objects which appear to be three dimensional. The computer continuously updates the perspective when the head is moved which generates this illusion.

Another possible client application is using the transmitter to move through a 3D cave. This application will be based off of a simple game where a ship is advanced through a cave and the walls get smaller as the game progresses.

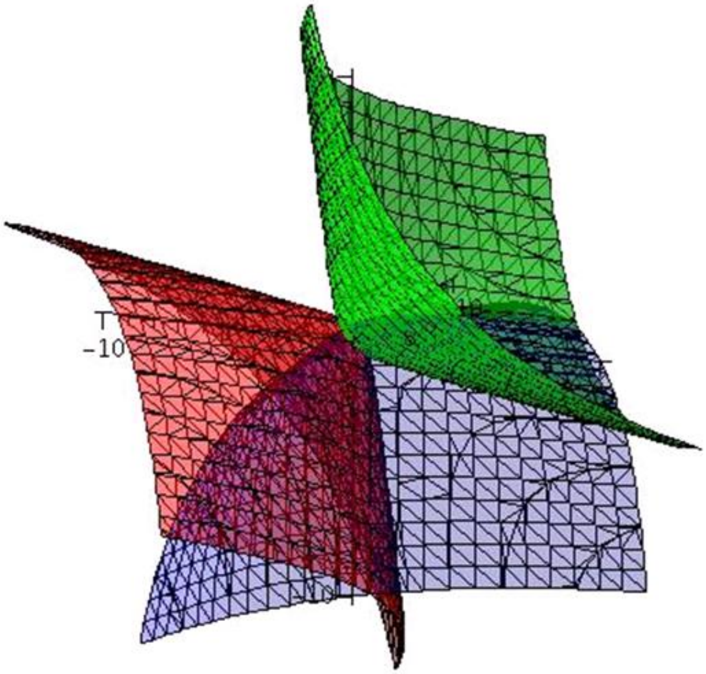


Figure 6. Example Multilateration Solution for a 3D System

Testing and Integration

Transmitter

Much of the testing of the transmitter system has already been completed. The first phase of testing was done using two ultrasonic transducers. The transmitting transducer was attached to waveform generator and the receiver was attached to the oscilloscope. The transmitter was secured to the table and the receiver was placed 6'9" away directly facing the transmitter. At this position no voltage was applied to the transmitter and the noise recorded by the receiver was measured (about 10-30 mV). Next, a square wave signal with a mid-range voltage of 5V at 40 kHz was applied to the transmitter. The corresponding signal, as seen by the receiver, was recorded. The previous step was repeated numerous times, moving the receiver to a new location and different angle in reference to the transmitter and the received signal was recorded. All recorded results were at least 500mV above noise levels, rendering the results acceptable, such that the design of the transmission system could continue.

The second phase of testing was also done using two ultrasonic transducers, except that the receiving transducer was attached to a designed filter and amplifier circuit to optimize the received signal. Doing the same tests as the previous phase, the signal was visually observed. It was determined that the range of the transmitter is at least as wide as the range tested above. This test may be repeated so constraints such as maximum distance and angles of variance can be analyzed.

The next phase of testing, which has not been completed, will follow the same outline as the previous phases, except the transmitting transducer will be attached to a designed power system which will utilize battery power. If, for any reason, any of the key parts of the system have to be redesigned, the previous tests will be performed again to verify proper functionality.

Receiver

Most of the receiver testing is done in conjunction with the transmitter testing outlined above. Each receiver is tested further by comparing the receiver outputs to each other receiving a signal from the same transmitter. The transmitter is pointed at the receivers and the signal is verified and compared after each op amp stage with an oscilloscope. If a signal is not comparable to the reference circuits it is adjusted by turning various trimpots in the circuit. Quick verification of proper circuit operation can be performed by simply checking if the receiver led light blinks when the transmitter is aimed in its direction.

Microcontroller

The microcontroller's purpose is to capture timer values and send those values to the computer. The first stage of testing will be to send four predetermined 32-bit integers and verify that the computer received those integers. These same four integers will be sent at 20Hz to ensure that the bandwidth and speed of the microcontroller will accommodate this frequency. The next stage will be to manually trigger the input capture pins on the board while software on the computer checks for data to be read on the port. All four input capture pins will be triggered at the same time and the computer and the four values will be verified that they are the same. Subsequent triggers will also be verified that they are increasing.

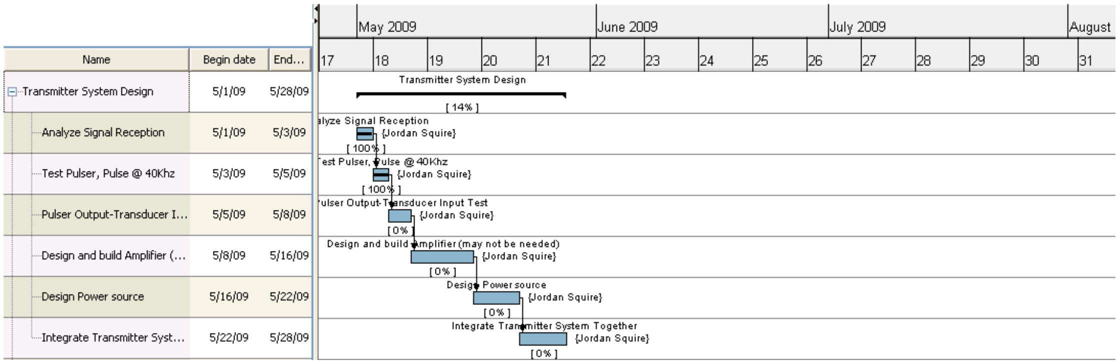
Multilateration

The values calculated from multilateration will be checked in a simulation and in real life. The expected time differences will be computed with a known transmitter location using the three-dimensional distance formula and the speed of sound, then these time differences will be ran through the multilateration program as if they came from the real receivers. The position and accuracy of the mathematics can be tested and experimented with to understand and fix any problems that may occur. Once the microcontroller and receivers are connected, the system is simply tested with a ruler. The transmitter is placed at a known location, and its location is measured with a rule. This measured value is then compared with the computed one from the actual timer values. Any errors will be easily discovered, and then corrected.

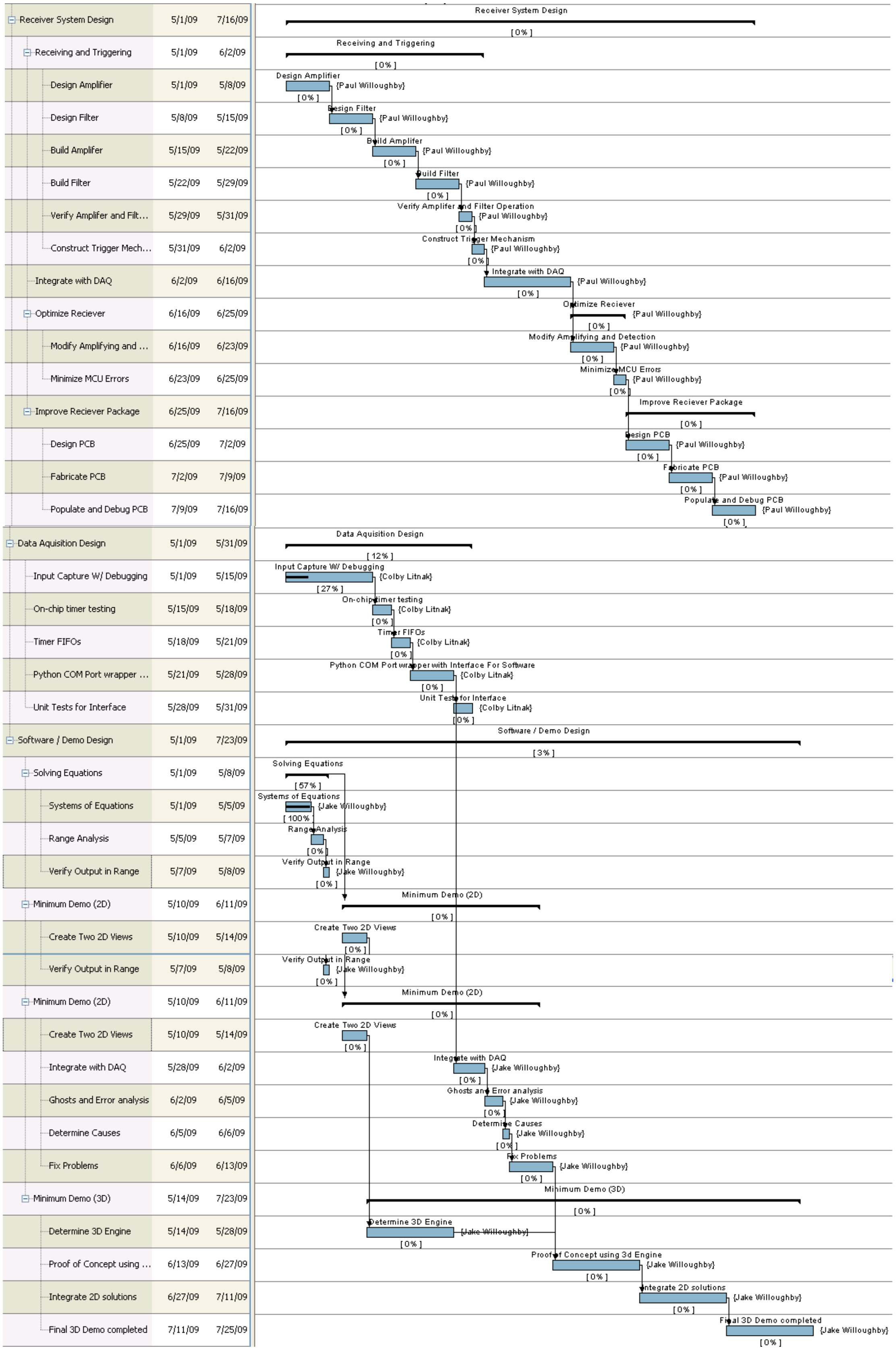
Group Communication Plan

The group holds meetings every Wednesday at 1:00 pm. At the beginning of each meeting an agenda is assembled where each member can add an item to be discussed in that meeting. Next, each member provides updates for the tasks which were assigned the previous meeting. The rest of the meeting follows the agenda. During the planning stages of the project, each meeting's agenda had many investigation tasks. It is anticipated that the majority of the remainder of the meetings will be used to discuss project issues and to solicit help from other members, less task assignments will be distributed during that time. Additionally, the team has a group mailing list and all frequently use instant messaging tools to communicate.

Schedule and Milestone



Name	Begin date	End...
Transmitter System Design	5/1/09	5/28/09
Analyze Signal Reception	5/1/09	5/3/09
Test Pulser, Pulse @ 40Khz	5/3/09	5/5/09
Pulser Output-Transducer I...	5/5/09	5/8/09
Design and build Amplifier (...	5/8/09	5/16/09
Design Power source	5/16/09	5/22/09
Integrate Transmitter Syst...	5/22/09	5/28/09



⊟ Receiver System Design	5/1/09	7/16/09
⊟ Receiving and Triggering	5/1/09	6/2/09
Design Amplifier	5/1/09	5/8/09
Design Filter	5/8/09	5/15/09
Build Amplifer	5/15/09	5/22/09
Build Filter	5/22/09	5/29/09
Verify Amplifer and Filt…	5/29/09	5/31/09
Construct Trigger Mech…	5/31/09	6/2/09
Integrate with DAQ	6/2/09	6/16/09
⊟ Optimize Reciever	6/16/09	6/25/09
Modify Amplifying and …	6/16/09	6/23/09
Minimize MCU Errors	6/23/09	6/25/09
⊟ Improve Reciever Package	6/25/09	7/16/09
Design PCB	6/25/09	7/2/09
Fabricate PCB	7/2/09	7/9/09
Populate and Debug PCB	7/9/09	7/16/09
⊟ Data Aquisition Design	5/1/09	5/31/09
Input Capture W/ Debugging	5/1/09	5/15/09
On-chip timer testing	5/15/09	5/18/09
Timer FIFOs	5/18/09	5/21/09
Python COM Port wrapper …	5/21/09	5/28/09
Unit Tests for Interface	5/28/09	5/31/09
⊟ Software / Demo Design	5/1/09	7/23/09
⊟ Solving Equations	5/1/09	5/8/09
Systems of Equations	5/1/09	5/5/09
Range Analysis	5/5/09	5/7/09
Verify Output in Range	5/7/09	5/8/09
⊟ Minimum Demo (2D)	5/10/09	6/11/09
Create Two 2D Views	5/10/09	5/14/09
Verify Output in Range	5/7/09	5/8/09
⊟ Minimum Demo (2D)	5/10/09	6/11/09
Create Two 2D Views	5/10/09	5/14/09
Integrate with DAQ	5/28/09	6/2/09
Ghosts and Error analysis	6/2/09	6/5/09
Determine Causes	6/5/09	6/6/09
Fix Problems	6/6/09	6/13/09
⊟ Minimum Demo (3D)	5/14/09	7/23/09
Determine 3D Engine	5/14/09	5/28/09
Proof of Concept using …	6/13/09	6/27/09
Integrate 2D solutions	6/27/09	7/11/09
Final 3D Demo completed	7/11/09	7/25/09

Figure 7. Gantt Schedule

Milestones: Components of the system will be functional at the following times:

Transmitter System and Data Acquisition: **May 30th**
Receiver System: **July 17th**
Software / Demo: **August 1st**

Risk Assessment

Risks in this project that may cause significant barriers to completion include the following:

- Too much noise received by the receiver transducer or can't power the transducer at full strength due to power supply limitation. There are three possible mitigation strategies including increasing the power of transmitted signal, reducing the angle between the transmitter and receiver, and reducing the distance between transmitter and receiver. A combination of these three strategies will likely be used if the need arises.
- Transmitted pulses ring for too long because of echoes. A possible mitigation for this risk is to increase the time between pulse batches.
- The system design expects that a single 40 kHz signal is the source of a trigger to capture the value of timer registers. The microcontroller latches each value and puts them into its corresponding FIFO. If a ghost signal causes one of the four inputs to latch unexpectedly, then one of the FIFO’s will have an extra value. If a signal fails to latch one of the inputs, then one FIFO will possibly be "behind". One way this risk will be mitigated will be to empty out the FIFO’s before sending values to the computer and only report the last values received and cause an empty of the FIFOs between pulses. This means that if the microcontroller falls behind in processing the FIFO’s for any reason, previously captured times may be lost.
- The USB COM port characteristic of this microcontroller introduces some bandwidth overhead which may be a bottleneck to the system. This interface supports sending up to 64 bytes at a time from the microcontroller to the computer. Testing has shown that if one byte is sent at a time, about 300 bytes/second is received. If 16 bytes (or four 32-bit integers) are sent a 8Kbytes/second throughput is achieved. The design specifies that one transmitter will pulse at 20Hz, which means the amount of data that will need to be sent is 20*(4 integers)*(4 Bytes per integer) = 320 bytes/second. If the pulse frequency is increased or more transmitters are added to the system, the bandwidth limitation may be reached. If more bandwidth is needed, the microcontroller can be configured to be a “true” USB device, and eliminate the overhead associated with it being recognized as a COM port. Microchip provides pre-built starter projects for the PIC32 chip with associated PC drivers for this configuration. It will take considerable more work, but may be necessary if the design needs to change.
- Input capture latch delays: One smaller concern is the case of two or more input captures registers requesting to latch at the exact same time, possibly causing one latch to wait for the other. This would introduce errors in the multilateration equations. It appears from all the documentation on the PIC32 that each input capture is built independent of each-other and do not lock the timer as a resource when latching its values.

Bill of Materials

Some materials will be required for this system. Table II contains the anticipated parts that will be needed to build this system.

TABLE II. BILL OF MATERIALS

Part	Quantity	Price	Supplier	Point of Contact
UBW32	1	39.95+shipping	SparkFun	Brian Schmalz (brian_schmalz@yahoo.com
Transducers	4	19.8	SparkFun	N/A
Transceiver	1	24.95	SparkFun	N/A
Headers	3	6.95	Sparkfun	N/A
PCB	4	11.98	Radio Shack	N/A

Conclusion

The proposed project is a three-dimensional object tracking system which will be able to identify the accurate 3D location of a tagged object in a room. An array of ultrasonic transducers (microphones) at a base station are tuned to pick up a specific high frequency, the inaudible sound waves coming from a transmitter on a trackable entity. The signal will pulse periodically allowing the base station to track the entity as it moves. Each microphone will receive a pulsed signal at a different time and by using the differences between these times of arrival and the known locations of the microphones in the array, the base station will be able to accurately calculate the location in three-dimensional space of the transmitting entity.

Bibliography

References

1. Kirkwood, Brent C. *Acoustic Source Localization Using Time-Delay Estimation.* M.S. thesis. 4 August 2003. Technical University of Denmark. http://brentkirkwood.com/science/documents/BCK_MS_Thesis_Final_20030811.pdf
2. Synchronous Counters - Final Report @ http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cwl3/report.html
3. Multilateration Example in C @ http://ralph.bucher.home.att.net/project.pdf
4. An Empirical Study of Collaborative Acoustic Source Localization @ http://nms.lcs.mit.edu/papers/ipsn07-girod.pdf

5. Locating Snipers using TDOA @ http://compilers.cs.ucla.edu/emsoft05/LedecziVolgyesiMarotiSimonBaloghNadasKusyDora05.pdf
6. Tutorial on how to get UBW32 up and going http://www.paintyourdragon.com/uc/ubw32/index.html
7. Microchip (makes of PIC32 Microcontroller on UBW32 board) @ http://www.microchip.com
8. UBW32 designer's website @ http://www.schmalzhaus.com/UBW32/
9. Page 1 contains a good starting point for implementing the receiver http://www.datasheetcatalog.org/datasheets/560/64108_DS.pdf
10. Johnny Lees Wiimote head tracking http://johnnylee.net/

Acknowledgments

1. Travis Stroud and his department at the University for allowing us to checkout a Spartan 3E FPGA and starter board.
2. Brian Schmalz for help getting some eagles schematic symbols for the UBW32 project board.
3. John Regehr for explaining input capture during March 3rd, 2009 lecture. Sparked a better idea of how to capture signal arrival times.