Underwater Acoustic Communication using Frequency Shift Keying with a BeagleBone Black

Chris Monfredo
Chris Johnson
Jon Holton
Greg Davis
Scott Hambleton

Guide: Les Moore
Customer: Andres Kwasinski

***Abstract*- This is the first iteration of underwater communication projects at RIT. An acoustic signal is sent between two BeagleBone Black single board computers for underwater autonomous robotic swarm applications.**

***Keywords*- BeagleBone Black, Acoustic, Printed Circuit Board, Parabolic Dish, Frequency Modulation**

I. Introduction

This project is the first underwater communications project sponsored by Boeing to promote aquatic engineering at RIT. This project used acoustic waves as the medium to transmit information. The full system was comprised of a microcontroller, the circuits that modulated and demodulated the signals, and the housings for the electronics which also served as the mounting for the speaker and hydrophone.

II. Modulation and Demodulation

Keeping the customer requirements in mind, a reliable 2-way communication scheme needed to be developed. The communication system had to be able to send and receive messages at 15 kbps, at a horizontal distance of 30 meters, and a vertical depth of 10 meters. The first design contrived, was a system that would use phase shift keying as the modulation scheme. After researching other systems that implemented a similar modulation scheme underwater, the confidence in the scheme was high.

What makes PSK a great modulation scheme is that the frequency component of the signal can be very high which would make achieving the data rate much easier. It was decided to use 4 phases, 45, 135, 225, and 315 degrees, to represent four different two bit binary numbers. A setup such as this allows for the phases to be gray coded such that an error on the demodulation side would most likely cause only a single bit error, allowing for a more efficient error correction scheme to be implemented.

There were two downsides to this modulation scheme. The first issue was the complexity of the circuitry needed to allow the system to reliably phase shift the signal. A number of chips were tested, but in the end a phase locked loop would have been needed to achieve the desired results. The second issue of the scheme was on the demodulation side. In order to determine which phase the incoming signal had, the demodulation and the modulation systems would both need to be referenced to the same zero phase. The logistics of trying to reliably obtain the same reference point was unfeasible.

After more research, frequency shift keying (FSK) was determined to be the next best alternative to reach our goals. The first design challenge came from our customer requirement that specified data rate. In order to hit the 15 kbps data rate, the carrier frequency of the scheme would need to be high enough such that the propagation delay due to water would not be a factor, but also be low enough that the speaker chosen to transmit the signal would not attenuate the amplitude of the signal too much. To help reach the 15kbps data rate, a simple 3rd party compression package was utilized. It can compress data at a 2:1 ratio with lossless compression, so a 15kb message gets reduced to a 7.5kb message.

A carrier frequency of 37.5 kHz was chosen based on the data rate calculations. Just as in the PSK system, four different gray coded frequencies were used to represent two bit binary numbers. The chosen frequencies were 29 kHz, 34 kHz, 39 kHz, and 45 kHz. These frequencies were chosen such that they would be above audible, but low enough that the speaker could reliably transmit the different frequencies, and such that their average frequency was equivalent to the carrier frequency.

At three periods transmitted at the carrier frequency, each symbol lasts for 85µs. Frame sizes were selected to be small enough to not be prone to many errors, but large enough to not cause issues with propagation delays. At 30m underwater, the propagation delay of an acoustic wave is 20ms. The data rate was determined by tabulating the propagation delay as a function of the number of overall frames being transmitted. A 15kb message, after compression and redundancy, and with all of its necessary control frames, was determined to be able to be sent in roughly 450ms, with a propagation delay of 400ms. This allows about 150ms for extra code overhead.

In order to produce the needed frequencies, a circuit was developed that could be easy to modify and maintain as the project progressed. It was determined through research that a sine wave could easily be generated by passing a square wave of the desired frequency through a simple band pass filter. To generate square waves at the four desired frequencies, a quadrature timer chip was used. By changing the resistor and capacitor values on each of the time pins, four different frequencies could be produced. In order to dial in the exact frequency values, and also have the flexibility to change the frequency values, potentiometers were used in place of standard resistors.

The next step in designing the modulation circuit was determining what type of band pass filter should be used. Due to space constraints on the board, as well as the desire to have a very sharp frequency response, an inductor-capacitor band pass filter was chosen over the standard resistor-capacitor band pass filter.

After each signal is passed through its specific band pass filter to filter out all noise produced by the chip except for the individual desired frequencies, the four signals were sent to a 4 to 1 analog multiplexer. This multiplexer was controlled via two control pins and one enable pin. This allowed the BeagleBone black to control which frequency was activated based on the message that was trying to be sent. Using a multiplexer allowed all four signals to be ready to be sent without any start up delay of the circuitry used to produce the signals. Although this delay seems minuscule, in a time sensitive operation such as this, the smallest amount of time makes a world of difference.

After the signal is modulated and sent through the water, the receiving circuitry needs to demodulate the incoming signal in order to extract the incoming information. Before demodulation is attempted, the signal is first passed through a band pass filter centered on 37.5 kHz and having an envelope of 27 kHz to 47 kHz. Since the incoming signal could be any of the four frequency values, a band pass filter such as this will remove any noise outside of the desired frequency range, while preserving the integrity of our signals being sent.

Extracting the information from the signal is done by determining the frequency values of the incoming signals. To perform this function, an FFT (Fast Fourier Transform) package was compiled and integrated with the receiver software module. Since the BeagleBone can only operate on discrete sets of data, an analog to digital converter was utilized to transform the incoming analog signal into discrete, digital samples, representing the signal. Through testing it was determined that an external ADC would need to be used due to the sluggish performance of the BeagleBone's on-board ADC.

Once the signal has been converted to digital values, an FFT was performed on every 64 bits of incoming data from the ADC. The frequency component containing the highest amplitude was taken as the frequency component that was sent, and the message was reconstructed on the receiving side of the system. Below is a MATLAB simulation to prove the concept out.

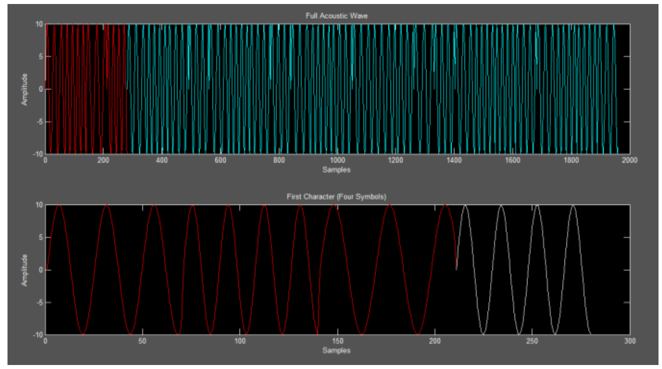


Figure 1: Acoustic Signal Output

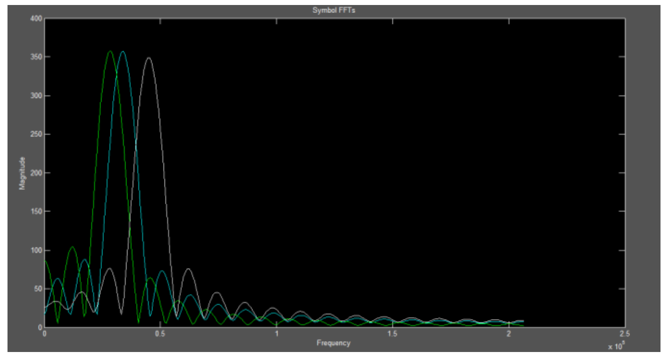


Figure 2: FFT Results from Signal

The primary problems encountered with the modulation and demodulation schemes were timing problems. On the modulation side, the multiplexer had issues with both startup noise and a startup delay preventing the output from being changed. Additionally, the sleep statements within the code used to hold the signal for the correct amount of time can sometimes be unreliable. The result, as observed on the receiver side, is that there is variance in how long each symbol's frequency is transmitted for, which made it nearly impossible for the receiver to read everything in correctly if it assumed the symbols changing every 64 samples. Slowing the data rate down and increasing the number of samples did not fix this issue either.

The problem was resolved with a slight redesign to the modulation and demodulation schemes. The 4-FSK modulation scheme was replaced with 2-FSK to ensure higher accuracy, and to allow for one of the unused symbols to be grounded. This grounded symbol is used as the neutral state for the multiplexer, so as to avoid startup noise from the multiplexer turning on. Additionally, the grounded symbol is sent after each symbol so that the receiver can detect that the end of a symbol has been reached. These changes were implemented as a state machine within the code, and the results were drastically improved. In the end the system was proven reliable at a data rate of 150 bps.

III. COMMUNICATION PROTOCOL

The communication protocol implemented by main control module is CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance). This protocol was selected due to its already wide-spread usage, its simplicity, and its ability to support many nodes within a network. A single node in the wireless network that has data to send will first remain quiet until the entire channel is quiet. It will then transmit a request-to-send (RTS) frame to its destination node, which will transmit back a clear-to-send (CTS) frame if it is not busy with anything else. The source node then transmits each frame of the message, and expects an acknowledgement (ACK) frame after each of them. If an acknowledgement is not returned, the frame is resent.

Each frame consists of two sentinel codes at the start and end of the frame. These are 8-bit codes that signify the start and end of a frame, so that the receiving unit understands that it didn't just read in random noise. The receiver reads in data symbol by symbol so long as an amplitude is present (or until the maximum number of symbols per frame is read). After error handling is performed, it simply checks that the first and last byes received are the sentinels. These effectively serve as an extra guard to ensure that the frame didn't just read in junk.

Frames also contain information indicating their type and redundancy. A frame has one of two types: control or message. Redundancy indicates whether the frame being transmitted is new or not. This is useful in a scenario such as the following: if device A transmits a message to device B, and device B transmits the acknowledgement, it's possible that device A never receives the acknowledgement. A then retransmits its message, but B has already read it in. The redundancy will be the same as before, so device B will know that it is a repeat frame and that it should be ignored.

IV. ERROR HANDLING

To combat the maximum allowed error rate on the channel of 10%, a hybrid scheme of forward error correction (FEC) and automatic repeat requests (ARQ) is used. The forward error correction takes each k-bit message and encodes it into an n-bit codeword. The receiver side utilizes the redundant bits included in the codeword to detect and correct any errors contained within the message. A 3rd party module utilizing Reed-Solomon codes was modified and integrated into the receiver and transmitter software modules. Reed-Solomon codes take a k-symbol message and add n-k redundant symbols in order to correct (n-k)/2 errors. To correct 10% of the errors in a message, 20% redundancy must be added. This would make a 201 symbol message into a 255 symbol codeword. In the event that the errors cannot all be corrected, an ARQ is invoked by simply doing nothing. By not processing the frame and thereby not sending an acknowledgement, the transmitter automatically repeats its transmission.

V. SOFTWARE ARCHITECTURE

Each major component within the software can be represented as a single module. The GPIO module initializes all of the necessary ports on the Beaglebone and provides memory addresses so that the ports can be accessed. This module is utilized by both the Receiver and Transmitter modules. The Transmitter module is utilized to create frames and control the multiplexer. The Receiver module reads information from the ADC, and passes digital data to the DSP module. The symbols returned from the FFT computations for the DSP module are passed through the Error Handler module to detect and correct any errors.

The controller module is the heart of the program and is where the communication protocol is implemented, which allows it to manage both the receiver and transmitter modules. A communicator module represents the user interface and holds the program's main method. When the program is started up, the communicator creates and initializes the controller module, which creates and initializes all of the other modules. The control module runs as its own thread and acts as a passive component, simply adhering to the communication protocol at all times. Input and output queues are used as a bridge between the user interface and the controller module. When the user sends a message, it is placed in the output queue, and sent as soon as the control module is ready to send it. When the control module receives messages, it places them in the input queue, which the user can manually check to see if any new messages have arrived.

VI. BEAGLEBONE BLACK VS. RASPBERRY PI

Various options for microcontrollers were considered based on factors such as cost, flexibility, and processor speed. As expansion potential was one of the major goals for this project, it was decided that a microcontroller with a GPP (General Purpose Processor) should be used instead of a dedicated DSP controller, as the GPP is much more flexible and easier to integrate with. Initially, the microcontroller selected was the Raspberry Pi, due to its versatility. The Raspberry Pi is widely-used for a variety of projects and has many support resources available online. It utilizes an ARMv6 processor with a 700 MHz clock, which was initially determined to be sufficient for reaching the data rate. Each Raspberry Pi is $35 so a pair of them easily fit into the budget.

After performing a more in-depth data rate analysis and determining that the timings would be very narrow, the Rasberry Pi was abandoned in favor of the Beaglebone Black. The Beaglebone is very similar to the Pi, but utilizes a newer ARM cortex-A8 processor with a 1 GHz clock. It features many more GPIO ports and integrates much more easily with electronics, and costs $45. The Raspberry Pi lends itself better to graphical applications, whereas the Beaglebone Black was determined to be much more suitable for our project.

VII. Power Systems

With all of the needed integrated chips and the processor known, the power systems can be designed. First, the maximum voltage range to power the speaker is needed. If 10W is desired to power the speaker, the RMS voltage can be found.

Given the speaker is a 4 ohm load, converting the RMS voltage to an amplitude results in an acoustic signal with amplitude 9V. Therefore, the minimum voltage range is +9V to -9V. In order to produce the negative voltages, two batteries are needed with the ground reference taken between them. Rounding up to the nearest standard battery value, 12V batteries were chosen.

In addition to the +12V and -12V, +5V and -5V are also needed to power the different integrated chips. In addition, the BeagleBone also needs 5V. Therefore, a DC-DC converter is needed. There were two options for a power converter: a linear regulator or a switching converter. A switching converter was chosen due to its much higher efficiency than a linear regulator.

Ripple voltage was a large concern for the 5V supplies as they would be powering the analog circuitry. This can be fixed by adjusting the capacitor on the output of the power converter. By adding a large capacitor, energy can be stored between switching cycles to reduce ripple voltage. However, high frequency oscillations in the voltage supply can still be a problem. To correct this, a second smaller capacitor can be added in parallel with the larger capacitor. This adds a second pole to the converter output which helps to attenuate any high frequency components.

To protect the circuitry, a fuse was added in series with both batteries. If either battery shorts, a fuse will blow and protect the other circuit components. In addition, the metal box was grounded to power ground which serves two purposes. First, if any connections short to the outside, the fuses will blow and remove power from the shorted connections. Second, by grounding the box, any electrical noise on the box can be removed.

Finally, small bypass capacitors were added near the power amplifier from +/-12V to ground. This helps to prevent high frequency noise on the power supply from contaminating the signal out of the power amplifier to the speaker.

VIII. Printed Circuit Board

While designing the circuit board there were several specifications that governed its design. Many of these specifications were due to requirements of the board manufacturer or physical constraints. First, the board's largest area could be twenty square inches. By having an area of twenty one square inches or less, a significantly cheaper ordering package could be used. Second, there was the number of layers the board would contain. The simplest board could use two metal layers. However, the large number of signal traces left no room for power traces. Therefore, a four layer board was designed to use the inner metal layers for power traces. Third was the thickness of the traces. The required thickness of trace is dependent on the current that passes through the trace. In order to ensure no problems with too thin of traces, all traces were oversized. The final widths used were 0.01" for signals, 0.02" for 5V power traces, and 0.04" for 12V power traces. Aside from the physical constraints, there were design constraints to get the desired performance out of the system. The two main concerns in this area were noise reduction and thermal management.

Noise can be a significant problem for the small analog signals of our systems. Several measures were taken to reduce possible noise in our signals. First, the circuit board consisted of three ground planes: power ground, analog ground, and digital ground. Each ground consisted of a metal fill plane that was connected to absolute ground (power ground) at a single point. A metal fill plane was used for each ground so that the resistance between any two points in the plane was as low as possible and therefore the entire ground plane was at the same voltage potential. This is also why metal fills were used for the power planes. By having a single point of contact between the grounds, this prevents digital or power switching current noise from flowing into the analog ground. In addition, the point of contact between the planes was designed to be an external jump. This allowed for inductors to be added between ground planes to remove additional noise if necessary.

Several steps were taken to avoid noise in the modulation circuitry. First, there was the removal of metal from the internal plans under the circuitry. A problem with using a four layer board is parasitic capacitance between layers which can cause noise. Sensitive circuits such as the multiplexer or LC shaping filters could pick up this noise and add it to the signals. Second, the inductors in the LC filters were placed perpendicular to one another. By placing them perpendicularly, this prevents coupling between the inductors which can change the performance characteristics of the filters destroying the actual signals.

Thermal management was the most important with respect to two components: the fuses and the power amplifier. Fuses work by the metal filament melting when it gets too hot due to high currents. If there is some accidental heat sinking effect on the fuses, the fuses may not blow until too high of a current. Therefore, the internal metal layers were removed underneath the fuses.

Thermal heat sinking was found to be incredibly important for the power amplifier. The amplifier has an average of 12W passing through it which causes the amplifier to heat up incredibly quickly. In addition, the IC has internal thermal protection which shuts of the chip. To prevent this, a significant amount of heat sinking was required. Our solution was heat sinking through the circuit board. The package of the amplifier, a T0-220-15, allows pins to be bent such that the chip can lay flat on the circuit board. In the area where the chip body would come in contact with the circuit board, a thermal metal fill plane was added. An array of small thermal vias was added in the metal fill plane to a second metal plane on the back of the circuit board. This second metal plane was connected to the mounting plate which was connected to the box. This series of metal connections allows the heat from the power amplifier to flow from the amplifier through the circuit board thermal vias to the mounting plate and eventually to the external metal box. This allows the heat to ultimately be dissipated into the water.

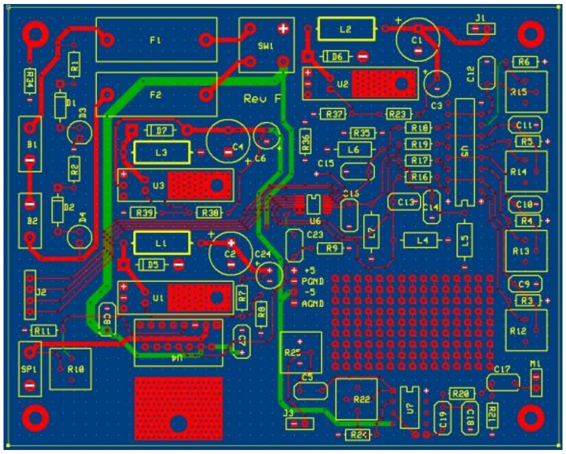


Figure (#): Printed Circuit Board Layout

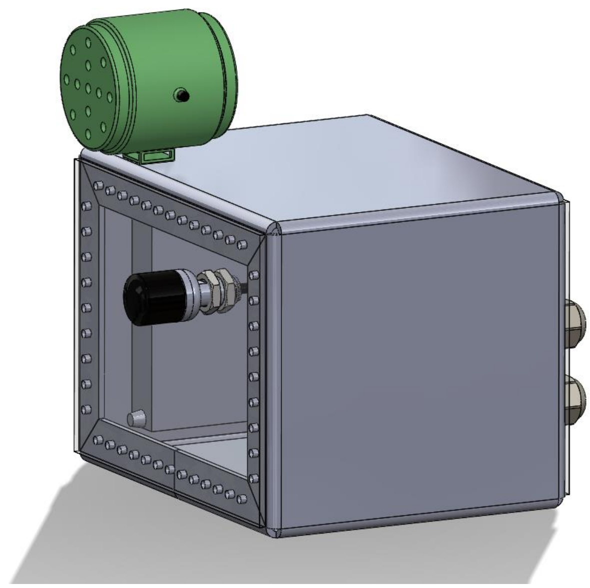
IX. Thermal Models

Two thermal models were created, one for each of the two housing designs. The first model was for the 3d printed housing, and expected the electronics to only reach a temperature of 63 degrees C with all 15 Watts of power leaving the system. The second thermal model was designed for the steel housing and looked more closely at the power amplifier and its position on the circuit board. Both thermal models for this system was designed with a steady state heat flow where the ambient external temperature was fixed to the highest temperature from our specifications and the heat flux was assumed to be the 12 Watts passing through the power amplifier. Due to the housing's small size and lack of circulation, the air was treated as a standing mass, so that all thermal resistances with the exception of the convection between the exterior of the housing and the water, was assumed to be conduction. As the conduction through air would be considerably worse than convection, this model was highly conservative. Several resistances that were included were the shape and length of the mounting plate as well as the PCB layers.

X. Housing

Initially the intended design was to 3d print a five-sided shell, 6"x8"x6", that would mount onto a single sheet of steel that would act as a hard mounting surface and interface with the water. The shell would have PEMs placed into the plastic and would feature a flange to place a gasket to keep the housing watertight. One of the greater concerns was how well the 3d printed shell would stand up to the pressure specification. The prime advantage would have been that the rapid prototyping system would have short lead time so that if damaged it could be replaced quickly. However that issue was removed when it became cost prohibitive to 3d print the shell as the material cost included the support structure the printer used to create the geometry of the flange, which was 6" off the base surface. This would lead to the that height, times the thickness of the flange, around the entire perimeter of the housing worth of extra material which drove the cost beyond our budget, especially with the electrical budget being of high concern at the time.

The second design was to bend sheet metal to form four of the sides and for the front and back to be plates that would mount to the housing via internal screws on one side, and via PEMS so that the system could be closed. These mounting plates would also feature the cable glands and mounting interfaces for the hydrophone dishes and speaker. The advantages of this design were that it had feasible cost and the system would be much more resistant to pressure, due to the ductile nature of steel. The down side was that the size of the box was fixed for future teams as well as that the waterproofing of the housing also depended on the quality of the welds that held it together.



Figure(#): CAD Model of the Housing

XI. Parabolic Dish

The minimum system requirements for the parabolic dish were a gain of 3 dB or more, both dishes couldn't have a combined cost exceeding $200, and due to the dimensions of the housing, the maximum radius was the width of the box. The equation for the parabola determined the distance the hydrophone would need to be to correctly amplify the signal. To prevent a focus either being impossible to place due to being either very far away or deep inside the dish, a shallow dish equation was selected. Once acquiring a relationship between the gain and the cross-sectional area of the dish, it was found that the design required very little area to meet the 3 dB goal. Therefore a function was derived that related the cost of material to the diameter which then computed the expected gain. This equation led to a final diameter around

7.5”. A major challenge was how to manufacture the parabola. With a lack of specific hardware for manufacturing a parabolic dish, 3d printing was considered. The cost of the dish was computed to be just underneath the allotted budget and the dishes were printed.

XII. RESULTS

As a system, the project was largely successful. Due to one of the ADC’s malfunctioning, the devices can only communicate in one direction, but fixing this problem should allow for 2-way communication. As a result of this limitation, the communication protocol is implemented but isolated, as it is dependent on each device being able to both send and receive data. The transmitter and receiver modules both incorporate the error handler and compressor modules, and messages can successfully be sent over a wired channel at 150 bps. This data rate is slower than the target data rate as there are both issues with the multiplexer being able to switch symbols quickly enough, as well as slow performance by the FFT package utilized by the DSP module.

The mechanical subsystem found some issues with leakage. The housing was tested overnight in a pool. As mentioned, there was minor leaking but the housing had a minor amount of water. There are ways to counter this in the future but there wasn’t enough time to implement them. The heat sink wasn’t tested so the thermal model wasn’t verified and due to issues of the speaker the parabolic dish was unable to be tested.

The circuit board was constructed and all subsystems were verified to function correctly. The modulation circuitry generated smooth, clean sine waves and the multiplexer will switch fast enough to hit the original data rate. The system power consumption was well within the maximum 15 watts. This allows the batteries to power the system for eight hours.

The testing of the hydrophone produced results very similar to the calculated values during system design. Therefore, theoretically the hydrophone should be sufficient even at the thirty meter maximum range even though this was not tested.

Unfortunately, problems were found with the frequency response of the underwater speaker. The corner frequency of the speaker was far below the operating frequency range of the system making it impossible to achieve acoustic communication using this hardware. For future revisions, it is suggested to use an acoustic transducer with a sensitivity of at least 150 dB and a corner frequency above 70 kHz.

XIII. CONCLUSION

In conclusion, not all the engineering specifications were achieved. However, a significant amount of progress towards a basic platform was achieved. Base circuitry and circuit board layout provide a template for future groups. The BeagleBone proved an adequate processor for this application. Although this was an ambitious project, much was accomplished.