BlueBuzz, an Open-Source Acoustic Modem

Scott Mayberry
Mechanical Engineering
Georgia Institute of Technology
Atlanta, Georgia
0000-0003-0822-4159

Jinzhi Cai
Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia
0000-0001-9426-5680

Fumin Zhang
Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia
0000-0003-0053-4224

Abstract—The high resource costs associated with performing aquatic-based research has led to the active development of mu-Net, a shared research infrastructure consisting of indoor and outdoor testbeds with open-source underwater networking software. These testbeds will require real-time acoustic communication, but there is a lack of commercial or other available acoustic modems that meet the needs for the indoor testbed. With the indoor testbed relying on custom hardware and an expected large variety of underwater robots, the acoustic modem must be: reconfigurable to fit in new ecosystems, open-source to permit modification, easy to access to allow researchers with varying hardware expertise to use/modify, functional in indoor environments with large multipath effects, and interoperable among a variety of hardware ecosystems. In this paper, we propose the BlueBuzz, an open-source acoustic modem that allows for easy integration into a variety of platforms, is reconfigurable, and has been experimentally verified in multiple environments

DOI: 10.1109/OCEANS47191.2022.9977326 acoustic modem, open-source, muNet, indoor testbed

I. INTRODUCTION

The expected doubling of the ocean economy between 2010–2030 relies on research-based innovations enabling industries such as offshore renewable energy, oil, and gas; marine aquaculture; and ocean monitoring, control and surveillance [1]. In particular, research infrastructure was identified as "one of the highest priorities of the ocean science and technology community" [2] in the November 2018 report by the US National Science & Technology Council.

While the excitement for aquatic-based research has attracted increasing attention from computing and engineering research communities, there is a high barrier to entry created by the need for specialized facilities, equipment, and training. Due to the high resource costs associated with performing aquatic research, shared research infrastructure has the potential to dramatically improve global research efforts. Researchers — under the umbrella μ Net project — have begun developing this shared research infrastructure, creating indoor and outdoor underwater robotic testbeds with open-source underwater networking software [3]—[5]. This shared research infrastructure enables a wide range of research areas — underwater robotic swarm control, underwater sensor networks, ocean monitoring and surveillance, smart marine aquaculture,

ONR N00014-19-1-2556 and N00014-19-1-2266; AFOSR FA9550-19-10283; NSF CNS-1828678, S&AS-1849228 CNS-2016582 and GCR-1934836; and NOAA NA16NOS0120028

978-1-6654-6809-1/22/\$31.00 ©2022 IEEE

etc. – and can assist in lowering the barrier to entry for aquaticbased research.

One of the core components needed to realize the aquatic μ Net testbeds is real-time acoustic communication. However, there exists a distinct lack of commercial or other available acoustic modems that meet the needs for the indoor testbed. The indoor testbed relies on custom robotic hardware to explore underwater controls, distributed data fusion, underwater sensor networks, and more. With expected rapid hardware changes and new robots being deployed, an acoustic modem for the indoor testbed must be:

- Reconfigurable: Different robots with varying chassis space for an acoustic modem are expected to be used in the indoor testbed, and the modem must be able to be physically redesigned to fit into these new ecosystems.
- Open-Source: Open-source software will allow for custom communication and localization strategies, while open-source hardware enables reconfigurability.
- Easy to access: A variety of researchers with varying hardware expertise will be working with the indoor testbed, and the acoustic modem must not require a high level of expertise to use/modify.
- Functional in indoor environments: The indoor environment is expected to have large multipath effects, and the acoustic modem must be able to communicate in such an environment.
- Interoperable: With the large variety of custom systems in the indoor testbed, any acoustic modem must be able to interface through either WiFi, Ethernet, USB, and/or UART.

Even though there is no available commercial or researchdeveloped acoustic modem that fits the needs of the indoor testbed, the aquatic community has developed modems that have focused on: communication speed, open-source, lowercost, and more.

For improved communication speed, research groups have explored orthogonal frequency division multiplexing (OFDM) for underwater acoustic (UWA) communication. In [6], a low complexity OFDM communication payload was designed for Micro-AUV to shore communication with a baud rate of 1.96kbps at 200m. In [7], while under a large time delay spread and in a time varying channel, a highly configurable OFDM



Fig. 1. Photo of BlueBuzz with paired transducer. View is of computer board. The analog board and amplifier board are stacked behind.

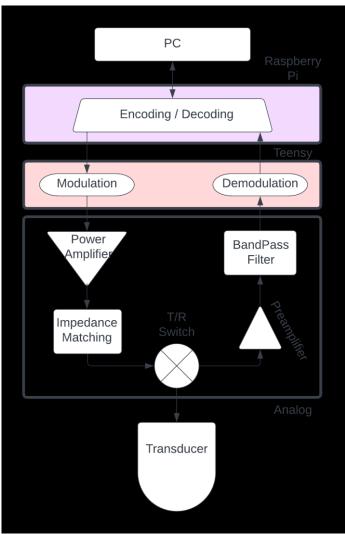
modem was used to achieve a maximum raw data rate of 4.39kbps at 0 bit error ratio (BER) after channel decoding. In [8], the significant bottlenecks in OFDM processing were identified, with OFDM block processing and synchronization improvements.

In the realm of open-source, researchers in [9] developed an open-source modem – named CoralCon – with a maximum bps of 1000 with an estimated BER of 10% demonstrated in open ocean. At the time of writing, the website hosting all designs for the modem was unavailable.

Research groups have also developed lower-cost modems. In [10], a lower-cost modem with a transducer was implemented at a price point of \$163 in 2022 USD with 200bps capability. In [11], another lower-cost modem was implemented with a tested range of 1m with a demonstrated baud rate of 300bps with 0 BER. In [12], a low-cost modem

named SeaModem was reported that could achieve a reported 750-2250bps.

None of the above modems meet the full needs of the robotics indoor testbed. To address this, we propose the BlueBuzz, an open-source acoustic modem that allows for easy integration into a variety of platforms, is reconfigurable, and has been experimentally verified. Explicitly, the main



contributions of this work are i) an open-source (hardware & software), reconfigurable acoustic modem capable of 250 bps at 0.5% BER using frequency hopping frequency shift keying (FHFSK) in an Olympic sized pool, ii) experimental results using the BlueBuzz in three environments: an acoustic tank, an Olympic sized pool, and a small lake, and iii) opensource interface modules (Ethernet and USB) for integration into future community developed modems to improve general interoperability.

II. SYSTEM HARDWARE

The BlueBuzz (Fig. 1) consists of an onboard computer for encoding/decoding, an onboard microcontroller (MCU)

Fig. 2. The BlueBuzz flow chart. For transmission, an external PC transmits a message to the onboard computer, where it is encoded. The MCU then modulates that message, and transmits through the transducer. For receiving, the transducer signal is amplified and demodulated by the MCU. The received

message is then decoded and error corrected by the onboard computer and transmitted to the external PC.



Fig. 3. The BlueBuzz acts as a transparent layer between the two communicating PCs with multiple interface options. This means that the two external PCs can transmit serial messages between themselves without any consideration of the physical layer of the modems.

for modulation/demodulation, an analog section containing transmission and receiving hardware, and a paired transducer as a transceiver (Fig. 2). Schematics and manufacturing files are located in the repository listed in the Appendix.

In operation, the modem is a transparent layer between two

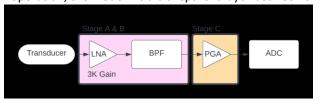


Fig. 4. The preamplifier has a 3kV/V fixed gain with a bandpass filter centered on 30kHz with a bandwidth of 35kHz. After the preamplification and band pass filtering, a final programmable gain amplifier of 1-7V/V boosts the signal to a range between 3-21kV/V (69-86db).

communicating nodes (see Fig. 3), and requires no special drivers to integrate into custom ecosystems.

A. Transmission

A 12-bit digital-to-analog converter (DAC) controlled by the MCU is used to generate modulated sine waves for transmission using either Frequency Shift Keying (FSK) or Frequency Hopping Frequency Shift Keying (FHFSK). The DAC output is fed into a custom amplifier with a 36db gain and an impedance matching circuit and is then transmitted by the transducer.

- 1) Amplifier: The amplifier system consists of two equivalent parallel amplifiers used to boost the transmission power. The amplifiers use two parallel TPA3116D2DAD chips and can provide up to 200W for transmission. However, only 30W is currently available due to the designed impedance matching system, but higher power transmission is capable without needing to change the amplifier system.
- 2) Impedance Matching: The impedance matching circuit consists of a passive matching circuit and a transformer. Driving the transducer efficiently requires both a matched impedance and high voltage.

To design the impedance matching circuit, both the impedance of the transducer and an off-the-shelf commercial

amplifier were measured using an impedance analyzer. The passive circuit was designed to roughly match the impedance of the transducer between 20-50kHz.

This process achieved a rough impedance matching, but higher voltage was also required. Using a variable transformer, the voltage was boosted, and the impedance matching circuit and maximum voltage levels were fine tuned. After a few iterations, a 1:13.71 transformer and passive components were selected (see Appendix for links to schematics).

B. Receiving

The preamplifier boosts the received signal heard by the transducer to a readable voltage range with three stages: a band pass filter with 100V/V gain (stage A), a band pass filter with 30V/V gain (stage B), and a selectable 1-7V/V programmable gain amplifier (stage C). The combined gain of all stages gives a selectable range between 3-21kV/V (69-86db) (see Fig. 4). The gain range of the modem was determined through experimentation.

The band pass filters of Stage A & B have a center frequency of 30kHz with a bandwidth of 35kHz. That places the 3db rolloff at about 17kHz and 52kHz. As the current BlueBuzz can only transmit between 23kHz-45kHz, the bandpass filters are used to remove extraneous noise.

- 1) Noise: In practice, the BlueBuzz has about 100-140mV of noise at 3kV/V gain. The ADC onboard can read a range of 0-4.096V at 0.001V accuracy, which gives a scale of relative signal vs noise capabilities (see Fig. 10).
- 2) ADC and Sampling: The MCU uses an analog-to-digital converter (ADC) to sample the incoming signal from the preamplifier. The ADC used is the LTC2315-12, a 12-bit ADC with an SPI interface capable of sampling at rates greater than 1MHz (the BlueBuzz is preconfigured to sample at 200kHz). The ease of use of an SPI enabled ADC, the high bit resolution, and the sampling speed make this a great IC to use for an open-source modem.

C. Computation

1) Low-level Microcontroller: The BlueBuzz uses a Teensy 4.0 by PJRC for modulation and demodulation. The Teensy 4.0 was chosen for a multitude of reasons: speed, capabilities, and the large community. The Teensy 4.0 has an ARM Cortex-M7 that can be overclocked to 812MHz without additional cooling, enabling real-time analog sampling and DSP without the need to transmit raw samples to a faster computer. The current BlueBuzz takes less that $1\mu s$ to sample the incoming signal from the preamplifier and then run a Goertzel algorithm with 50 samples to detect the edge of an incoming signal.

The Teensy community is also quiet large, supported by electrical engineers, roboticists, and mechatronics engineers all with the aim of optimizing this low-cost, high-performance device. For an open-source platform such as the BlueBuzz,

having a large support community for each major computation center was a necessity.

- 2) Onboard Computer: The onboard computer is a Raspberry Pi Zero W. This device encodes messages for/decodes messages from the onboard microcontroller (MCU). The Pi Zero W is not currently capable of running DSP code fast enough for the current modem configuration without major optimization, and thus DSP is handled on the MCU.
- *3)* Interfaces: The BlueBuzz attempts to minimize integration difficulties by providing multiple interface options: WiFi, Ethernet, and USB/UART. See Appendix for repository links for the Ethernet and USB modules used in the BlueBuzz.
 - 1) WiFi: The onboard computer, a Raspberry Pi Zero W, comes pre-equipped with WiFi, and can be used to access the device remotely.
 - 2) Ethernet: The onboard computer has been extended to include an integrated Ethernet port with a programmable MAC address. This allows easy integration into systems that are networked, and also allows for multiple modems to be added within a system without duplicate MAC addresses. The speed is limited to ≤4Mb/s.
 - 3) USB/UART: An added USB micro-B port allows for communication between a PC and the modem using UART. It should be noted that power is not provided to or through this port.

D. Power & Energy Usage

In receive mode, the BlueBuzz uses about 2.2W. The receive circuitry uses 0.5W, the Teensy uses 0.5W, and the Raspberry Pi Zero W uses 1W. The rest is lost to power conversion. In transmit mode, the BlueBuzz can use anywhere between 2.4W to 30W (50W has been seen but not for continuous use). Power out is controlled by reducing/increasing the amplitude from the modulating DAC.

For input, the BlueBuzz accepts any voltage between 1216.8V, which is the standard voltage range of a 4S LiPo battery.

No testing has been done to measure the true conversion of power to sound, and some power is bound to be converted to heat.

III. DIGITAL SIGNAL PROCESSING

Digital signal processing (DSP) is done on the onboard MCU, and not the onboard computer. The current onboard computer, a Raspberry Pi Zero W, cannot process FFT or Goertzel fast enough for a 200kHz sampling rate without optimization, and thus is only used for error correction and encoding.

A. MCU vs FPGA

Many modems today utilize FPGAs for DSP with great success [13], [14]. However, FPGAs require a higher level of expertise than the ubiquitous microcontroller (MCU), and a decision to use an MCU was made based on a focus to enable easy modification of the BlueBuzz.

B. Frequency Identification

The MCU uses the Goertzel algorithm for frequency identification. While FFT is more standard, Goertzel has many benefits that are preferable when looking at a limited number of frequencies.

- Goertzel can be run iteratively, requiring less instantaneous computation power before the next sample is to be taken. This reduces the required computation abilities of the MCU while allowing for high speed sampling.
- Goertzel is more efficient than FFT for a low number of frequencies. As the pertinent frequencies are currently limited to two different frequencies to represent the two possible incoming symbols, Goertzel is significantly faster.
- Significant parts of the Goertzel algorithm can be precomputed, further decreasing the need for real-time computation.

By adopting Goertzel instead of FFT, the BlueBuzz significantly reduces hardware requirements while providing decent communication capabilities.

C. Signal Edge Detection

Detecting the edge of an acoustic signal is difficult. Due to the mechanical nature of a transducer, it takes time to drive the transducer to the proper volume. This time is measured in milliseconds, but is enough that the leading edge of the arriving signal is significantly reduced in volume compared to the actual message. In order to account for this, two strategies are used: sliding window and parallel messages.

The sliding window uses a one-in one-out strategy for incoming samples. The newest sample is added to a recorded list of samples, and the oldest sample is dropped. The MCU then runs a Goertzel algorithm on the list of samples in realtime to detect if a message has started. This "message started threshold" is determined by the magnitude of the expected frequency. The BlueBuzz always transmits the largest value symbol as the first symbol, and thus the receiving MCU always knows which frequency it should be looking for.

One difficulty with the sliding window strategy is tuning the "message started threshold" to always line up with the incoming acoustic signal. To solve this, the MCU solves parallel messages spaced apart by a set number of samples. For example, there are 9 thresholds that can be triggered to start a separate message thread in the current configuration. The arriving signal will increase in magnitude in the frequency domain, triggering any number of the message threads between 1 and all 9 thresholds. Upon message completion, all

active message threads will be transmitted to the onboard computer for error correction and decoding. For clarity, this means that up to 9 messages will be transmitted to the onboard computer for error correction and decoding, with the most likely message being the one transmitted to the external PC.

D. Message Terminus Identification

The message terminus is detected by a drop in magnitude of all the pertinent frequencies. This causes an overshoot in the number of symbols received as it takes time for all frequencies to drop below the relevant thresholds due to the multipath effect in the indoor environment and the mechanical nature of the transducer. The overshoot is fixed by requiring messages

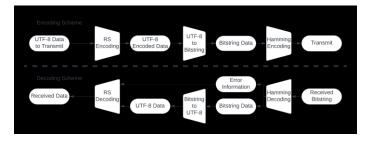


Fig. 5. The BlueBuzz encodes with a inner/outer structure of Reed-Solomon and Hamming.

to be of a length equal to a multiple of a preselected fixed length, with all extra symbols dropped.

E. Error Encoding & Decoding

While the MCU handles DSP, the onboard computer handles encoding/decoding and error correction (Fig. 5).

For transmission, a message from an external PC is converted to a UTF-8 formatted string and then is encoded using a Reed-Solomon code. The message is then converted to a bitstring, where the message is encoded a second time with a Hamming code. This forms a two layer encoder with the inner being Reed-Solomon and the outer being Hamming. The dual encoded bitstring is sent to the microcontroller,

where it is modulated using the chosen communication scheme (FSK, FHFSK, etc.).

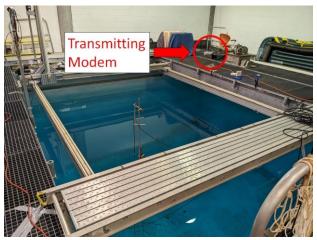
The dual Reed-Solomon and Hamming encoder is quite flexible, and can be modified to fit the expected noise level of the channel. A fixed message size is used, but this message size can be changed easily before deployment. The Reed-Solomon encoder can be modified for a large range of errors or erasures, with the cost of increased error correction capability being less real data sent per second. The Hamming size can also be modified.

For receiving, the symbols from the MCU are transmitted to the onboard computer. The outer layer Hamming decoder splits the message into preselected sizes and fixes single bit

errors and records double bit errors for each submessage. The Hamming corrected bitstring submessages are recombined and converted to a UTF-8 string. The Reed-Solomon decoder fixes the erasures (known error positions reported by the double bit errors from the Hamming decoder) and errors (unknown error positions). If the message has too many errors to decode, the modem will drop the message, otherwise the complete message is transmitted to the external PC.

IV. EXPERIMENTS & ANALYSIS

A. Acoustic Tank



The Acoustic Water Tank located on Georgia Tech's main campus is 12m long, 6.5m wide and 7m deep. Acoustic tests performed in the acoustic tank were done at a 1m depth along the diagonal of the tank at varying distances and varying baud rates (see Fig. 6 for experimental view and Fig. 7 for BER vs Baud Rate).

Both BFSK and FHFSK were tested in the tank. For BFSK, the BER rapidly increased after a baud rate of 50. Further experimentation determined that this was due to the multipath effect in the small indoor environment. These BFSK experiments inspired the FHFSK implementation to try to improve the BlueBuzz's baud rate in an indoor environment. As can been seen in Fig. 7, FHFSK has a higher communication rate capability compared to BFSK in the acoustic tank: 200bps at 2% BER vs 50bps at 0% BER.

Fig. 6. GT Acoustic Water Tank test with BlueBuzz in box in corner. The transmitting transducer is out of view underneath the bulkhead, but is about 1m deep. Picture taken from receiver point.

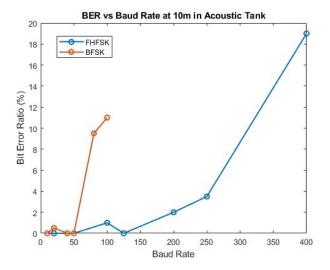
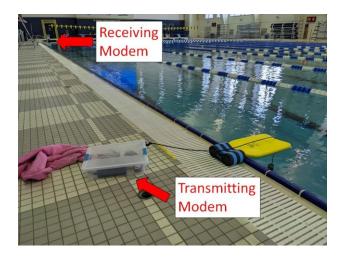


Fig. 7. BER vs Baud Rate at 10m distance at the CRC pool for FHFSK and BFSK. Four messages of 100 symbols each were transmitted at each baud rate for BFSK and FHFSK.



B. 50m Olympic Pool (CRC Pool at GT)

At the Georgia Tech CRC Pool, two modems were used to communicate messages of 100 symbols in length at varying baud rates and varying distances. Fig. 8 shows the experimental setup with the transmitting modem being moved along the side of the 50m pool while the receiver stayed stationary. Fig. 9 shows the BER vs baud rate at the maximum distance in the pool (50m) using FHFSK. Compared to the FHFSK results from the acoustic tank (Fig. 7), the larger pool environment had better performance. This supports the argument that the multipath effect was the major contributor to the increased BER in the acoustic tank. With minor error encoding, the BlueBuzz can transmit at rates of 250bps in an Olympic size pool, and 200bps in smaller pools.

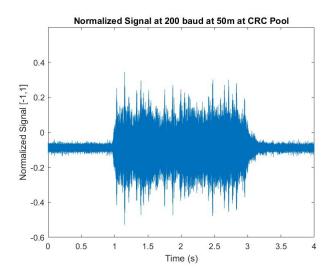
Fig. 8. 35m communication test at the CRC pool at Georgia Tech. Tests were performed at 3, 8, 15, 25, 35, and 50m.

FHFSK BER vs Baud Rate at 50m distance at the CRC pool. Four messages of 400 symbols each were transmitted at each baud rate.

C. Lake Test at Kraken Springs

At Kraken Springs the goal was to determine the transmission distance capabilities of the BlueBuzz using FHFSK. Tests were done at a distance of 220m (Fig. 11 and Fig. 12) at a baud rate of 125, and had an average BER of 38%. This high BER was caused by the lower transmission power of FHFSK compared to BFSK. Due to the impedance curve of the BlueBuzz transducer, some of the chosen frequencies selected in the FHFSK scheme cannot be transmitted with high power (impedance mismatch at those frequencies). The frequencies used in the FHFSK scheme were chosen for the indoor 50m tests, but were not impedance matched well enough to properly transmit 220m.

The transmission power limitation for FHFSK implies that there are two communication zones for the BlueBuzz: short-range/higher-speed (FHFSK) and long-range/lowerspeed (BFSK). However, the BlueBuzz is aimed at indoor



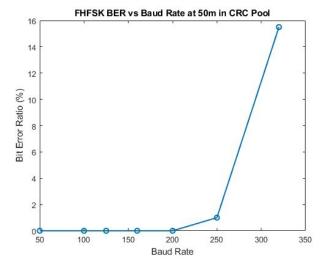


Fig. 9.

Fig. 10. An example of the received signal relative to noise at 50m in an

Fig. 11. Kraken Springs Test Site Aerial View. The red line represents the 220m distance between the receiver and the transmitter. The blue circle represents an estimated possible 100m of communication using FHFSK.

environments, so FHFSK performance can safely be used.

V. CONCLUSION

The BlueBuzz is a fully open-source (hardware & software) acoustic modem aimed at indoor testbeds. It can, in indoor environments with severe multipath effects, communicate at 250 bps at 0.5% BER using FHFSK and at 50 bps at 0% BER using BFSK. The BlueBuzz is interoperable and can be reconfigured to fit in a multitude of ecosystems.

ACKNOWLEDGMENT

The authors express their thanks to Dr. Francois Guillot for extensive access to the Georgia Tech Acoustic Water Tank. The authors also express their thanks to the Georgia Tech Research

Olympic sized pool.

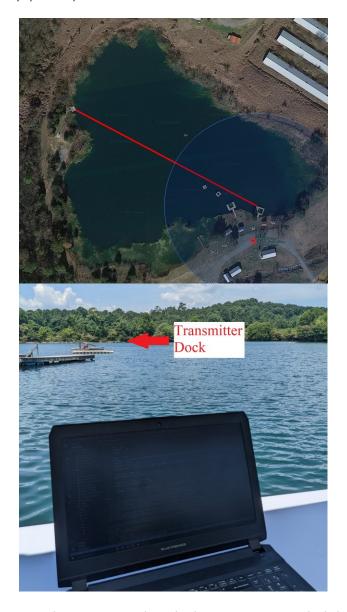


Fig. 12. Kraken Springs Testing Photo. The BlueBuzz transmitter is on the dock 220m away. The picture is taken from the receiving dock.

Institute (GTRI) for invitation to and usage of Kraken Springs during their reserved time.

REFERENCES

- OECD, The Ocean Economy in 2030. Paris: OECD Publishing,
 4 2016. [Online]. Available: https://www.oecd-ilibrary.org/economics/the-ocean-economy-in-2030 9789264251724-en
- [2] The National Science & Technology Council, "Science and Technology for America's Oceans: A Decadal Vision,"
 - https://oeab.noaa.gov/wp-content/uploads/2020/Documents/ Science-and-Technology-for-Americas-Oceans-A-Decadal-Vision.pdf,
 - November 2018, accessed: Feb 19, 2019.
- [3] S. Mayberry, J. Wang, Q. Tao, F. Zhang, A. Song, X. Hong, S. Dong, C. Webb, D. Dugaev, and Z. Peng, "First Step Towards μ Net: Open-Access Aquatic Testbeds and Robotic Ecosystem," in *The 15th International Conference on Underwater Networks & Systems*. New York, NY, USA: ACM, 11 2021, pp. 1–8. [Online]. Available:

https://dl.acm.org/doi/10.1145/3491315.3491322

- [4] A. Song, F. Zhang, Z. Peng, X. Hong, S. Mayberry, J. Wang, Q. Tao, S. Dong, C. Webb, D. Dugaev, and et al., "Mu-net: Shared infrastructure for mobile underwater networks." [Online]. Available: https://aquatictech.org/
- [5] A. Song, Χ. Hong, F Zhang, Peng, and 7. Wang, "Mu-Net: Community-shared infrastructure for mobile underwater acoustic networks," The Journal of the Acoustical Society of America, vol. 150, no. A197-A198, 10 2021. [Online]. https://asa.scitation.org/doi/10.1121/10.0008115
- [6] W. Jiang, Q. Tao, J. Yao, F. Tong, and F. Zhang, "R&D of a low-complexity OFDM acoustic communication payload for Micro-AUV in confined space," EURASIP Journal on Advances in Signal Processing, vol. 2022, no. 1, p. 64, 12

2022. [Online]. Available: https://asp-

eurasipjournals.springeropen.com/ articles/10.1186/s13634-022-00898-9

- [7] Y. Zhou and F. Tong, "Research and Development of a Highly Reconfigurable OFDM MODEM for Shallow Water Acoustic Communication," *IEEE Access*, vol. 7, pp. 123569–123582, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8809753/
- [8] H. Yan, S. Zhou, Z. J. Shi, and B. Li, "A DSP implementation of OFDM acoustic modem," in *Proceedings of the second workshop on Underwater networks WuWNet '07*. New York, New York, USA: ACM Press, 2007, p. 89. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1287812.1287831
- [9] A. A. Sheikh, E. Felemban, and A. Ashraf, "Coralcon: An open source low-cost modem for underwater IoT applications," in 2017 13th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP). IEEE, 9 2017, pp. 503–508. [Online]. Available: http://ieeexplore.ieee.org/document/8117054/
- [10] B. Benson, Y. Li, B. Faunce, K. Domond, D. Kimball, C. Schurgers, and R. Kastner, "Design of a low-cost underwater acoustic modem," *IEEE Embedded Systems Letters*, vol. 2, no. 3, pp. 58–61, 9 2010.
- [11] M. Y. I. Zia, P. Otero, and J. Poncela, "Design of a Low-Cost Modem for Short-Range Underwater Acoustic Communications," Wireless Personal Communications, vol. 101, no. 1, pp. 375–390, 7 2018.
- [12] G. Cario, A. Casavola, M. Lupia, and C. Rosace, "SeaModem: A lowcost underwater acoustic modem for shallow water communication," in OCEANS 2015 - Genova. IEEE, 5 2015, pp. 1–6. [Online]. Available: http://ieeexplore.ieee.org/document/7271721/
- [13] N. Nowsheen, C. Benson, and M. Frater, "Design of a high frequency FPGA acoustic modem for underwater communication," in OCEANS'10 IEEE SYDNEY. IEEE, 5 2010, pp. 1–6. [Online].

Available: http://ieeexplore.ieee.org/document/5603819/

[14] L. Xu and S. Yan, "Design of Underwater Acoustic Modems through High Performance DSPs," in *Proceedings of the International Conference on Underwater Networks & Systems - WUWNET '14*. New York, New York, USA: ACM Press, 11 2014, pp. 1–2. [Online].

Available: http://dl.acm.org/citation.cfm?doid=2671490.2674598

APPENDIX

Repositories for schematics and manufacturing files:

- BlueBuzz: https://github.com/scottmayberry/BlueBuzz
- Ethernet Module: https://github.com/scottmayberry/ enc28j60 ethernet module
- USB/UART Module: https://github.com/scottmayberry/ usb to ttl module
- USBHub Module:

https://github.com/scottmayberry/ usb-hub-module