Team 1904 – Design Review

Enhancing Software Defined Radios for Underwater Acoustic Modem

Sponsor: The MITRE Corporation
Faculty Advisor: Dr. Peter Willett
Team Members: Hunter Malboeuf (EE), Davis Meissner (EE), Greg Palmer (CMPE)
Project Goal

• Develop an underwater acoustic communication system using two Software Defined Radios (operation in a 5 gallon bucket environment)

• Initial goal is one-way SDR communication system

• Stretch goal #1 is a two-way SDR real-time communication system

• Stretch goal #2 is to incorporate equalization and a convolutional coding scheme

• Project is comprised of three phases:
  - Simulation
  - Hardware Integration
  - Analysis of Received Waveform
Background

• Over 70% of the earth is covered by water
• The ocean is a 3 dimensional space
• 11,000 meters at the deepest point
• Only 2-3% is explored
Underwater Communications

• Manned Vehicles
  - small research submarines
  - large military platforms
• Unmanned Vehicles
  - Autonomous Underwater Vehicles
  - Remotely Operated Vehicles
  - Hybrid Underwater vehicles
Why Acoustic?

- **Radio Frequencies (~1m range)**
  - Absorbed by seawater

- **Light (~100m range)**
  - Strong dependence on water clarity

- **Ultra Low Frequency RF (~100 km)**
  - Massive antennas (miles long)
  - Not practical outside of Governments

- **Cables**
  - Expensive & impractical for mobile units
Underwater Communication Challenges

- Multipath effects – transmitted messages bounce off the sea surface and bottom, arriving at the receiver at different points in time
- Power losses over the path depend on water temperature and depth of operation for the transmitter and receiver
- Doppler spreading due to transmitter and receiver motion

*Controlled environment of this project allows for AWGN channel to approximate these effects*
Project Requirements

• Transmit and Receive small text messages in underwater environment
  - Using ASCII encoding for text, but potential to extend to other types of data (.jpg, etc…)

• Waveform Development
  - Modulate using differential phase shift keying (DPSK)
  - Error correction to compensate for errors caused by channel
  - Interleaving to redistribute bits across waveform
  - Synchronization between transmitter and receiver to determine start of message signal.
Project Components

• Two Ettus X310 Software Defined Radios
• Two acoustic transceivers (hydrophones)
• Host machine to interface between radios:
  - Linux PC
  - Two embedded processors (Udoo X86)

*Components are MITRE provided
Communication Block Diagram

Signal Source → Compression → Error Control Coding → Permutation/Interleave → Modulation → Frequency Shaping → Carrier Shift

Received Signal → De-compression → Decoding → Reverse Permutation/Interleave → Bit Decision → Matched Filtering → Carrier Removal

- only if time permits
- complete
- to be completed
Compression

- Encodes information into fewer bits than the original message
- Source encoding will be done before our message signal is sent
- Source decoding is applied after the signal is received
- Lempel Ziv possibility
  - may not be of much benefit for our small data and ASCII messages

010100111001010011010

010100111001010011010

Diagram:

Compression — Error Control Coding — Permutation/Interleave — Modulation — Frequency Shaping — Carrier Shift
Error Detection/Correction

• Detect & correct errors that occurred during transmission

• Hamming Code (7, 4)

• Might Add cyclic redundancy parity check to check validity of overall message
Interleaving

To mitigate against sporadic bursts of noise, interleaving is used

Original data example: 0xABC123
Interleaved example: 0xA1B2C3

This way corrupted bits are more often able to be recovered from the Hamming encoding scheme
Modulation

- Varying our waveform with the information in our message by modulating our carrier signal
- Currently using Binary Phase Shift Keying (BPSK)
- Will work to implement a noncoherent Differential Phase Shift Keying (DPSK) modulation scheme

[2] Example of DPSK from tutorialspoint.com
Frequency Shaping

- Root Raised Cosine Filter
- Current use in our code was provided by MITRE
- Still need to research more about this

[3] Frequency response of raised cosine filters
wikipedia.com
Carrier Shift

• Stretching or compression of waveforms in transmission
• Doppler effect from transmitter and receiver moving
Synchronization

• Transmitter and receiver need to establish a synchronized clock to be able to properly interpret any incoming messages

• Transmitter will transmit a chirp signal that is gradually increasing in frequency that the receiver can lock onto and be ready to receive message in sync with transmitter
Simulation - Completed

• MITRE provided starter MATLAB acoustic communication system tool
• MATLAB Tool has been modified to:
  - Transmit and receive over AWGN channel
  - Read data from a text file
  - Encode data to ASCII characters to bits
  - Apply Hamming(7,4) Code single bit error correction
  - Use matrix interleaving for bit redistribution before transmission
  - Calculate BER and plot BER vs. SNR
MATLAB Graphs
Simulation - Planned

• Planned additions to MATLAB model include:
  - Replacing current BPSK modulation scheme with DPSK
  - Adding a cyclical redundancy parity check
  - Synchronization using preamble “chirp” signal
Hardware Integration

• Hardware will be acquired from MITRE on November 19th
• MATLAB waveform developments will be implemented in C++
• Integration of acoustic transceivers and SDRs will be done in C++
• Group will decide on Linux PC vs. embedded processors
Analysis of received waveform

• Relevant quantities that will be extracted from data include:
  - Bit error rate (BER)
  - Signal to noise ratio (SNR)
  - Actual data transmission and reception rates
• Lab data will be compared to theoretical results from MATLAB
Fall 2018 Semester Schedule

9/21/18  9/28/18  10/5/18  10/12/18  10/19/18  10/26/18  11/2/18  11/9/18  11/16/18  11/23/18  11/30/18  12/7/18

- Add FOR loop to transmit data
- Plot BER vs. SNR
- Read data from text file
- Encode data to 8 bit ASCII
- Matrix Interleaving / De-interleaving
- Hamming Error Correction (Single Bit)
- Design Review Preparation
- Proposal
- Replace BPSK with DPSK
- Add CRC check
- Add Tx-Rx synchronization
- Explore hardware integration techniques
Questions?
Works Cited

[1] MITRE. 2019 Senior Design Project Outline
[2] DPSK Image from Tutorials Point
[4] Doppler Effect Image