E156 Final Project: Frequency-Modulated Binary Acoustic Communication System

Eyassu Shimelis[1]

***Abstract*— The purpose of this project was to design an audio communication system capable of encoding and transmitting binary data. The only constraint was that each transmission session must last under thirty seconds. This report details the design, implementation, and test conducted for a frequency-modulated binary communication system.**

I. INTRODUCTION

Most of today's communication is composed of the transmission of digital data using electromagnetic waves (telecommunication). Systems such as WiFi, Bluetooth, and LTE are examples of communication standards used in most of our current communication systems. Most of these standards operate in mega- (10^6 Hz) or gigahertz (10^9 Hz) frequency ranges. This makes it possible to transmit vasts amounts of information in very short time spans.

Unlike electromagnetic communication systems, audio communication (sonic com) transmits data through sound. For the purposes of this final project, we were tasked to transmit audio through a laptop speaker. This reduces the available transmission frequency down to the kilohertz range (10^3 Hz), several orders of magnitude slower than their electromagnetic counterparts.

Most communication systems can be simplified into three subsystems: the transmitter, the channel, and the receiver (see Fig. 1). The transmitter is the entire electromechanical system responsible for encoding, modulating, and transmitting messages through some respective medium (sound, electromagnetic radiation, etc.). The channel is the model of the 'space' between the transmitter and receiver. This encompasses the medium, added and convoluted noise, physical obstacles, and any other environmental elements that may distort the signal. The receiver is the final component of the system, it is responsible for receiving transmitted signals, decoding them in the presence of noise and distortion, and finally outputting the predicted message.

II. DATA

The data being transmitted is a 1.3 Mb Reuters markup file containing approximately 1.25 million characters. Using the ASCII format, this is equivalent to almost 10 million bits. Transferring this much data in the span of 30 seconds would require a communication system capable of transmitting over 300 kilobits per second. A simple feat for modern telecommunication systems, but not for acoustic communication.

[1] Eyassu Shimelis is an Engineering student at Harvey Mudd College in Claremont, CA. `eshimelis@hmc.edu`

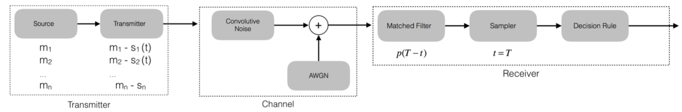

Fig. 1. Abstracted communication system diagram: transmitter system, receiver system, and intermediary channel model.

III. SYSTEM DESIGN

A. Transmitter

The transmitter, in this system, converts a text file into a vector of binary American Standard Code for Information Interchange (ASCII) values. The bits are modulated onto a carrier wave, and the transmitter outputs the resulting audio file. The system is a simple Binary Symmetric Channel that uses frequency modulation to send a 1 or a 0 (see Fig. 2).

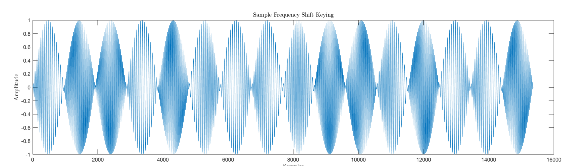

Fig. 2. Sample FSK waveform encoding binary data (0110100001101001). Each pulse represents a single bit of information.

Each bit was encoded as the following pulse-shaped sine waves:

$$s_0(t) = A\sin(2\pi f_0 t)\sin(\pi t/T_s)$$
$$s_1(t) = A\sin(2\pi f_1 t)\sin(\pi t/T_s),$$

where $f_0 = 800$ Hz and $f_1 = 1500$ Hz are the respective frequencies of each bit, and $\sin(\pi t/T_s)$ was added to shape the pulse in order to reduce rapid changes in volume. The frequencies were selectively chosen to reside in the audible frequency range, since computer speakers are designed to operate well in this range. In order to improve the results of the matched filter in the receiver, the two frequencies are also selected so that one is not an integer multiple of the other.

IV. SYNCHRONIZATION

Unlike most communications systems, all of the decoding occurs offline after all of the data has been collected. Since the receiver applies a matched filter to decode the waveforms, it is very important that the receiver can distinguish the start of the communicated message. In order to achieve to synchronized the transmitter with the receiver, a short chirp was added to the start of the transmission (see Fig. 3, top figure). A chirp is a sine wave with changing frequency. More specifically, the chirp used in this projects last two seconds, and sweeps from 2000 Hz to 30 Hz and back to 2000 Hz. The chirp serves the purpose of finding and accounting for the arbitrary delay between the start of the recording and the message itself.

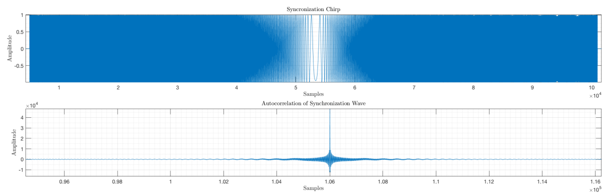

Fig. 3. Top: Syncronizing chirp, sweeps from 2000 Hz, to 30 Hz, to 2000Hz. Bottom: Correlation of synchronizing chirp with itself.

The chirp waveform was chosen for one significant property: its autocorrelation function falls off extremely quickly (see 3, bottom figure). This phenomenon is due to the non-uniform frequency of the waveform. By adding this chirp to the start of a transmission, it is possible to find the delay in samples by convolving the chirp with the recorded audio. The final transmitted waveform is shown in Fig. 4.

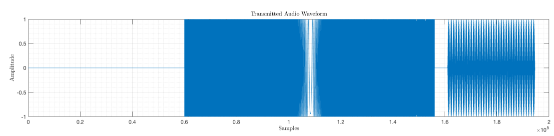

Fig. 4. Sample transmitted waveform with synchronizing chip. The FSK encoded binary data follows the synchronizing chirp, which leads the start of every transmission.

V. Channel

As discussed earlier, the channel is the representation of the signal distortion between the transmitter and the receiver. For the purposes of our model, we assume the channel has two components: a low-pass filter to represent the laptop speaker system and Additive White Gaussian Noise (AWGN) to represent the random noise in the room, such as from the AC system. Since the laptop speaker acts as a low-pass filter, it adds convolutive noise. AWGN, on the other hand, is added, zero-mean, normally distributed noise.

VI. Receiver

The receiver used in this project is a matched filter receiver. The process of matched filtering involves convolving known signals (in this case, $s_0(t)$ and $s_1(t)$) every bit period, and selecting the signal with the highest correlation. The resulting signal is sampled at every time T to create a vector of predicted messages. This process, however, is highly sensitive to misalignment and is discussed in following sections.

VII. Error Correction

In order to improve the accuracy of transmission, the system employs a (7,4) cyclic code. This type of error code effectively increases the distance between different messages and can actively correct for small bit errors at the receiver. This encoding was chosen to minimize the amount of redundancy since each transmission was only limited to 30 seconds. Systematic cyclic encoding adds parity checking to the end of binary blocks. These parity checks increase the relative distance of error between successive binary blocks. In the case of small bit errors, the parity check bits will not match the original set of characters, and a computed syndrome can actively correct the message to the closest (or most probable) message.

VIII. Results

A. Simulation

Before conducting real-world tests, the system was tested using a simulated channel in Matlab. The channel consisted of only AWGN because a spectrum analysis of the transmitted signal revealed that it did not contain any significant higher frequency components, which meant that a low-pass filter would not affect the resulting signal.

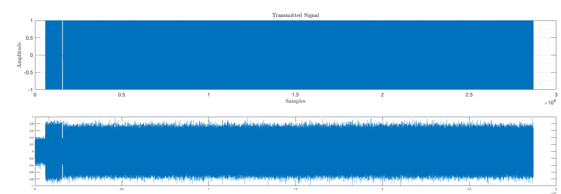

Fig. 5. Transmitted signal before and after transmission through the simulated AWGN channel.

As shown in Fig. 5, the simulated channel adds White Gaussian Noise to the transmitted signal. After correlating the transmitted signal with the synchronization wave, the received signal is aligned and ready to be sent through the matched filter. Fig. 6

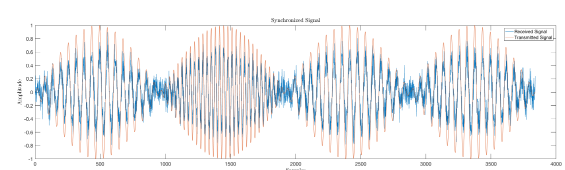

Fig. 6. Received signal aligned with transmitted signal after correlation with synchronization wave.

The simulation worked without any error for signal to noise ratios as low as 15 dB.

B. Experimental

Unlike the simulated results, the final experiment was not as successful. Despite manual fine-tuning, the resulting output was incorrect. The system bit error hovered around 50%.

IX. Future Improvements

After some thorough consideration, there are some possible sources of error that may contribute to an unsuccessful receiver. First, reducing the transmission period per bit would improve the alignment process, especially if done manually. Another possible improvement would be increasing the sample rate of the transmitted signal, otherwise, small errors will accumulate in the matched filter, resulting in erroneous predicted messages. An additional issue with the experiment we conducted is the possibility that recording the message on a phone resulted in an audio file with a mismatched bit period. Although the first few periods may line up, small errors will propagate for longer files or transmissions with shorter bit periods.