Design and Implementation of Orthogonal Frequency Division Multiplexing (OFDM) Signaling

Design Proposal

Study by:
Alan C. Brooks
Stephen J. Hoelzer

Advisors:
Dr. In Soo Ahn
Dr. Thomas L. Stewart

EE 451
December 8, 2000

Project Summary

This project will focus on Orthogonal Frequency Division Multiplexing (OFDM) research, simulation, and implementation. OFDM is a modulation technique especially suitable for wireless communication due to its resistance to inter-symbol interference (ISI). After researching OFDM, simulation in MATLAB will be completed. The main part of this project will be using the simulation results as a guide to implement OFDM on a DSP board. Depending on the DSP hardware used, the MATLAB code will need to be converted to either C or Simulink. This makes our code compatible with the software tool for that particular DSP board. To test the DSP code, we will verify that the input and output vectors of the MATLAB simulation and DSP implementation correspond.

Importance

With the rapid growth of digital wireless communication in recent years, the need for high-speed mobile data transmission has increased. New modulation techniques are being implemented to keep up with the desire more communication capacity. Processing power has increased to a point where OFDM has become feasible and economical. Since many wireless communication systems being developed use OFDM, it is a worthwhile research topic. Some examples of current applications using OFDM include DSL, DAB (Digital Audio Broadcasting), HDTV broadcasting, IEEE 802.11 (wireless networking standard).

Project Description

This project consists of MATLAB simulation and DSP implementation. Using a DSP design tool such as dSpace and Simulink or TMSC60 tools, a real time OFDM transmitter will be built. Figure 1 shows a simplified flowchart of the MATLAB simulation code.

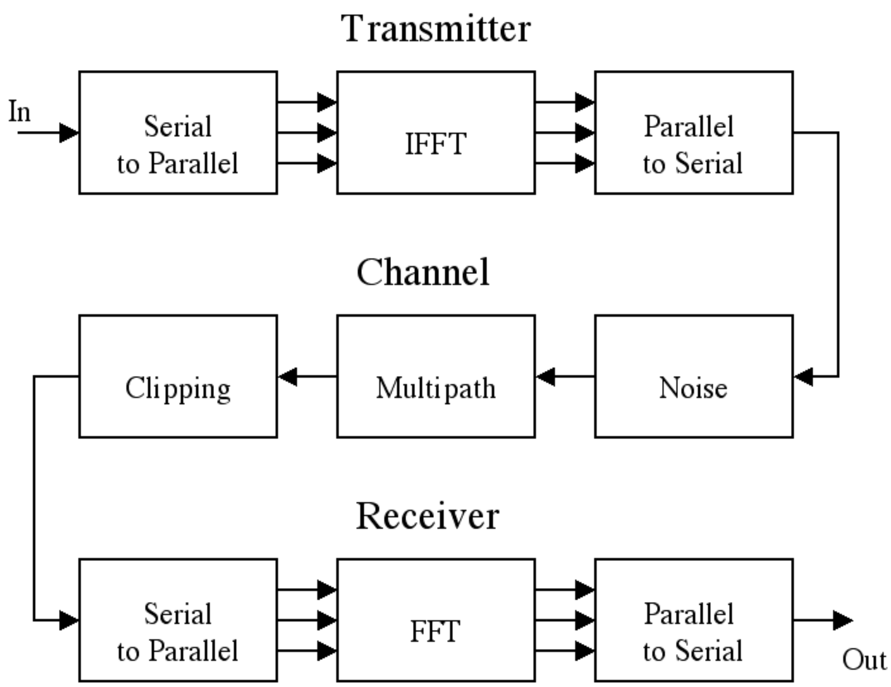

Figure 1: OFDM Simulation Flowchart

The transmitter first converts the input data from a serial stream to parallel sets. Each set of data contains one symbol, S_i, for each subcarrier. For example, a set of four data would be [S_0 S_1 S_2 S_3].

Before performing the Inverse Fast Fourier Transform (IFFT), this example data set is arranged on the horizontal axis in the frequency domain as shown in Figure 2. This symmetrical arrangement about the vertical axis is necessary for using the IFFT to manipulate this data.

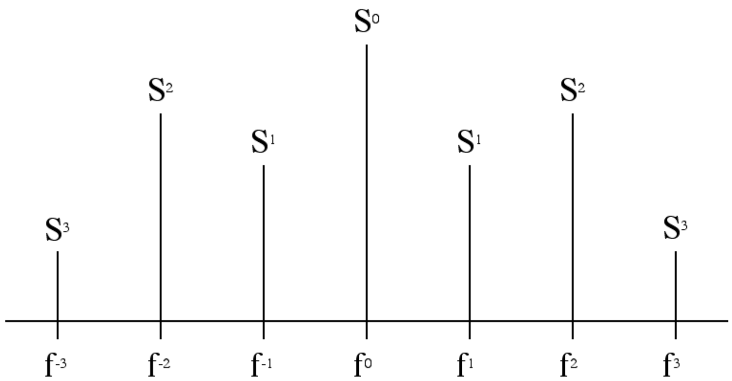

Figure 2: Frequency Domain Distribution of Symbols

An inverse Fourier transform converts the frequency domain data set into samples of the corresponding time domain representation of this data. Specifically, the IFFT is useful for OFDM because it generates samples of a waveform with orthogonal frequency components.

Then, the parallel to serial block creates the OFDM signal by sequentially outputting the time domain samples.

The channel simulation will allow examination of the effects of noise, multipath, and clipping. By adding random data to the transmitted signal, simple noise can be simulated. Multipath simulation involves adding attenuated and delayed copies of the transmitted signal to the original. This simulates the problem in wireless communication when the signal propagates on many paths. For example, a receiver may see a signal via a direct path as well as a path that bounces off a building. Finally, clipping simulates the problem of amplifier saturation. This addresses a practical implementation problem in OFDM where the peak to average power ratio is high.

The receiver performs the inverse of the transmitter. First, the OFDM data are split from a serial stream into parallel sets. The Fast Fourier Transform (FFT) converts the time domain samples back into a frequency domain representation. The magnitudes of the frequency components correspond to the original data. Finally, the parallel to serial block converts this parallel data into a serial stream to recover the original input data.

The current version of the MATLAB simulation accepts binary, text, or sound as input. It then generates the corresponding OFDM transmission according to variable setup parameters.

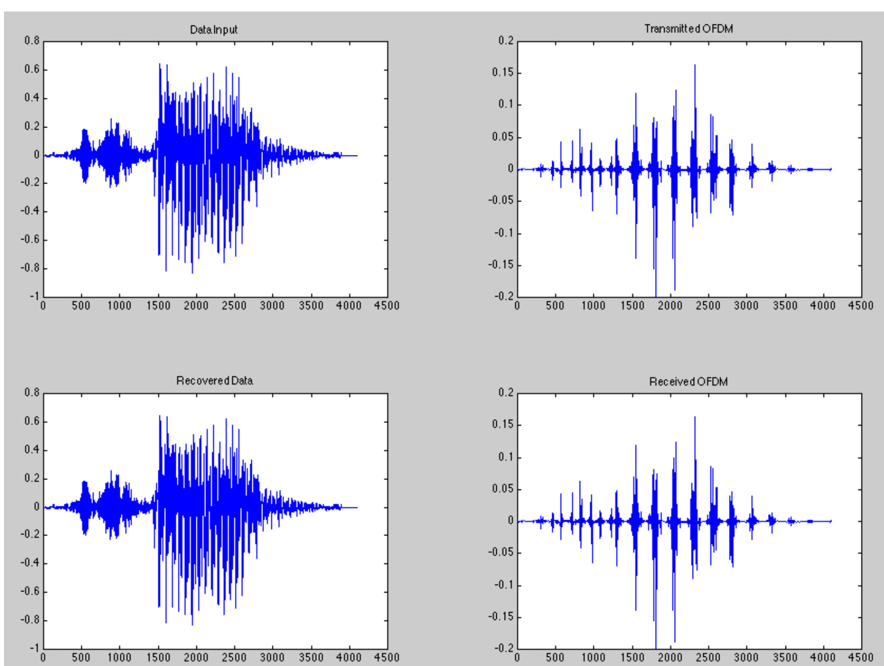

Figure 3: MATLAB Simulation with Sound Input

Figure 3 shows one example of the graphs generated by the current MATLAB code. The upper left plot is the input sound file. After modulation, the corresponding OFDM transmission is shown on the upper right. For this example, a perfect channel was assumed which means that the received signal (lower right) exactly matches the transmitted. Finally, the received OFDM signal is demodulated to reproduce the original input data, shown in the lower left plot.

Schedule

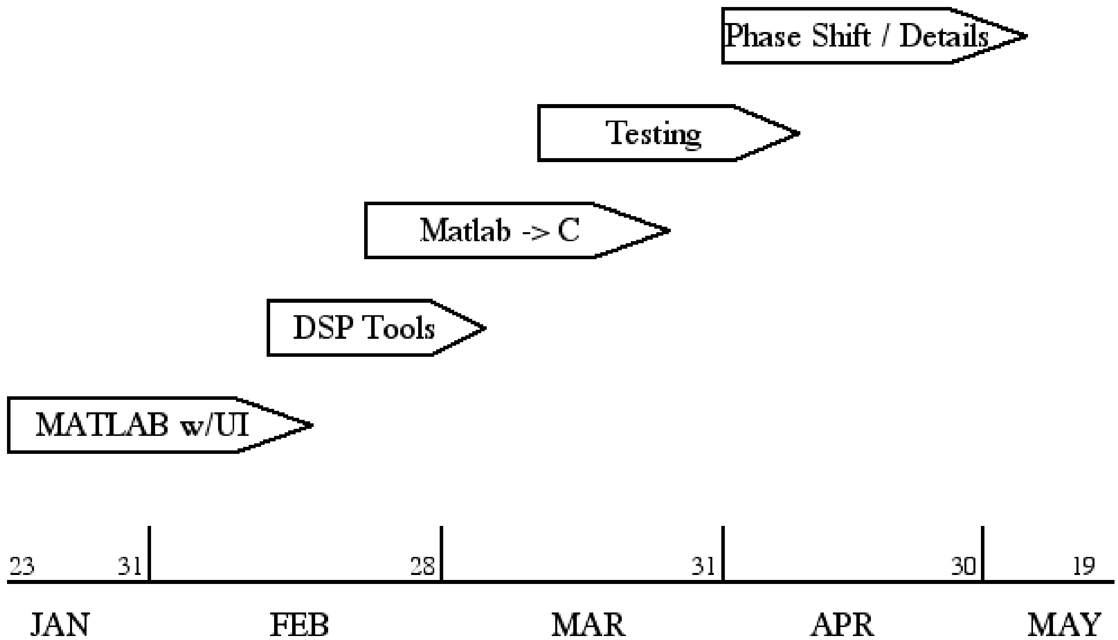

Figure 4: Spring 2001 Schedule

As shown in Figure 4, the main goals to accomplish in Spring 2001 include finalizing MATLAB, generating the corresponding DSP code, and then testing the system. The overlap between tasks shows that in a two engineer team, parallel work paths will be carried out at the same time.

In detail, MATLAB code will be improved to make a hardware implementation feasible. This will involve adding QAM and guard periods, as well as windowing, interleaving, and coding features. The next phase will be to learn the DSP tools. By writing simple test simulations, the functionality of the DSP tools will be explored. Once these tool are understood, the MATLAB code will be translated into the appropriate format for the DSP board. Simulink can be used with the dSpace board, but the TMSC60 board uses C. The testing phase will involve verifying that the DSP board produces results consistent with the MATLAB code. Also, once the OFDM system works correctly, the characteristics of the OFDM transmission will be examined. This may include phase shift and synchronization techniques.

Bibliography

[1] Van Nee, Richard, and Prasad, Ramjee. *OFDM for Wireless Multimedia Communications*. Boston: Artech House, 2000.

[2] O'Leary, Seamus. *Understanding Digital Terrestrial Broadcasting*. Massachusetts: Artech House, 2000.

[3] Bahai, Ahmad R. S., and Saltzberg, Burton R. *Multi-Carrier Digital Comunications:Theory and Applications of OFDM*. New York: Kluwer Academic/Plenum Publishers, 1999.

[4] Keller, Thomas, and Hanzo, Lajos. "Adaptive Multicarrier Modulation: A Convenient Framework for Time-Frequency Processing in Wireless Communications." *IEEE Proceedings of the IEEE* 88 (2000): 609-640

[5] Lawrey, Eric. OFDM Wireless Technology. 11 May 2000. 7 Nov. 2000. http://www.eng.jcu.edu.au/eric/thesis/Thesis.htm

Patent History

Class/Subclass

370	Multiplex Communications
370/203	Generalized Orthogonal or Special Mathematical Techniques
370/208	Particular set of orthogonal functions (subset of 203)
708	Electrical Computers: Arithmetic Processing and Calculating
708/400	Transform (subset of 200)
708/403	Fourier (subset of 400)
708/404	Fast Fourier Transform (subset of 403)

Historical

3,488,4555	Orthogonal Frequency Division Multiplexing (Jan 6, 1970)

Current

<u>370/208</u>

6,125,124	Synchronization and sampling frequency in an apparatus receiving OFDM modulated transmissions (Sept 26, 2000)
6,021,110	OFDM timing and frequency recovery system (Feb 1, 2000)
5,694,389	OFDM transmission/reception system and transmitting/receiving apparatus (Dec 2, 1997)

<u>708/404</u>

6,115,728	Fast fourier transforming apparatus and method, variable bit reverse circuit, inverse fast fourier transforming apparatus and method, and OFDM receiver and transmitter (Sept 5, 2000)

Equipment List

Hardware: Computer, DSP Board (dSpace or TMSC60)

Software: Matlab, dSpace Tools (Simulink), C60 Tools (C)