THE MARCONI PROJECT

www.sgsolutions.ca/marconi

Transparent Wireless Modem

Block Verification

May 13th, 2005

Department of Electrical and Computer Engineering
University of Waterloo
200 University Ave. W., Waterloo, ON

Aaron Cheung
Hardware Lead
00141344

Stefan Janhunen
Algorithms Lead
00057388

Vincent G. Liu
Marketing Lead
20069140

Shu Wu
Software Lead
20069140

Table of Contents

1. Hardware Layout

There are four main blocks of hardware that form the radio modem. Shown in Figure 1, they are the serial interface, the digital signal processor (DSP), the audio codec, and the audio interface. Each sub block serves a specific function in the wireless modem. The following sections will briefly describe the functions of each sub block, followed by an overview to evaluate whether the according specifications listed in the project specification document can be satisfied.

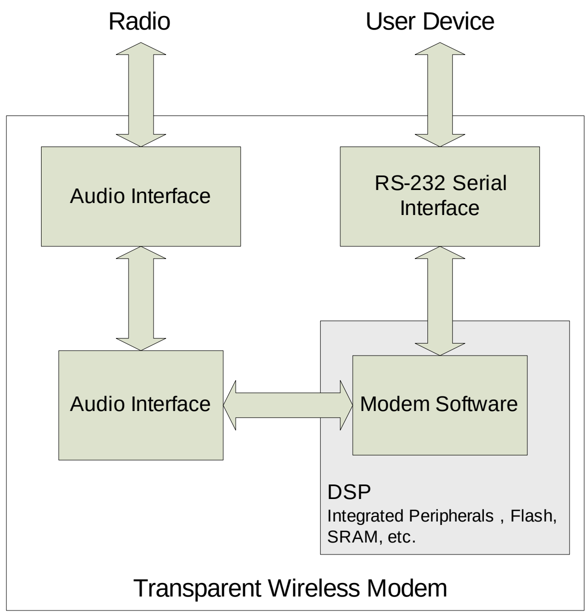

Figure 1 - Hardware Blocks Abstraction

1.1 Serial Interface

The main function of the serial interface is to allow data transfer between the radio modem and the device attached, a portable computer for example. Since the radios in consideration will support half duplex data transfers, one of the two modems will be transmitting data while the other will be receiving data at any given time. On the transmission side, the serial interface on the radio modem receives information from the device and feeds the information to the DSP for processing, and the reverse operations are being carried out on the receiving side. There are three major concerns regarding the serial interface, namely the speed of data transfer capability, number of serial ports, and choice of industry standard.

In order to ensure the final product operates as a transparent modem, the serial interface must be able to transmit and receive data above a minimum speed. Referring from the project specification document, it is required that the modem should be able to transmit or receive data greater than a rate or 100 bits per second (bps) over a reasonable channel.

Since the norm of serial data transfer speed with current technologies can easily exceed 250kbps, the appropriate choice of hardware component can satisfy this requirement.

Regarding the number of serial ports, the radio modem requires only one serial port under normal operating conditions. However, during the development phase, it is much more convenient for the designer to have an additional serial port for hardware debugging. Consequently, it is desirable to have a serial interface that has two serial ports. Since there are serial interfaces that support multi-channel data transfers with multiple receivers and drivers, this requirement can be satisfied by selecting the appropriate part.

As for the choice of industry standard, it is required that an appropriate serial interface standard that is widely used by the industry be selected, such as the RS-232 interface. According to the RS-232 standards, it is assumed that the distance between the two devices is less than 50ft. As well, the data transfer rate must be fairly low. Since the radio modem satisfies both of these assumptions, it is possible to adapt the RS-232 standards with an appropriate choice of component.

1.2 Digital Signal Processor (DSP)

The DSP block is the most important block within the hardware section. Its main function is to carry out most, if not all computation required for the radio modem to function properly. On the transmission end, the DSP receives binary information from the serial interface. The DSP then performs numerous operations on the data in preparation for transmission, and finally sent off the bit stream to the audio codec. Operations carried out by the transmitter side DSP include framing, encoding, and modulation, and the reversed order of operations takes place at the receiver's end. Of all the operations performed by the transmitter and receiver's DSP, the demodulation operation at the receiver's end's DSP is the most computational power intensive. As such, it is necessary to investigate the computational speed of the DSP. In order to ensure proper operation, it is also necessary to ensure the DSP has sufficient memory.

During demodulation on the receiver's end, the DSP block receives Pulse Code Modulated (PCM) bit stream from the audio codec. The DSP then transforms the data back to the original form and sends it to the serial interface. Assuming the simplest modulation-demodulation scheme where a four tone Frequency Shift Keying (FSK) method is used, the DSP will need to identify which of the four possible frequencies are received. In order to do so, the DSP needs to perform four sets of convolution on the bit stream quickly. The act of convolution can be broken down into multiplication and accumulate, or MAC in short. Since the radio modem must transmit or receive no less than 100bps, the DSP block must be able to perform a lot of MAC operations quickly. At current technologies, DSPs are sufficiently powerful to provide the required number of MAC operations per sample. For instance, the TMS320F2811 DSP from Texas Instruments is capable of performing one MAC operation per cycle at a maximum frequency of 135MHz, meaning at the maximum of 135 million MAC operations per sample. If the audio input from the radio modules are sampled at 8kSa/sec, and assuming no other operations are being performed at the time of demodulation, the DSP can perform at a maximum of $\frac{135\times10^{6}\,MAC/\sec}{8kSa/\sec}$ =16.9kMAC/Sa, which is more than sufficient. Consequently, if an appropriate DSP that is capable to carry out a sufficient number of MAC operations per sample, then it is possible to satisfy the computational requirement for the radio modem.

Besides the concern on the DSP's computational power, it is also necessary to consider the DSP's peripherals included with the package. For the DSP to perform computations on the data for transfer there must be sufficient FLASH and RAM for software storage and efficient computations, respectively. The TI TMS320F2811 DSP is capable of supporting up to or more than 36K of RAM and 256K of FLASH memory, which is more than sufficient for the purpose of this radio modem. Besides this TI part, most DSP in the current market has internal memory integrated within the package, which simplifies system integration tremendously.

1.3 Audio Codec

The audio codec serves as a digital to analog converter (DAC) on the transmitting side and an analog to digital converter (ADC) on the receiving side of the radio modem. In essence, the audio codec interfaces between the audio interface and the DSP block. On the transmitter side, the audio code receives PCM bit stream from the DSP and output an analog signal to the audio interface, and the reversed operations are carried out on the receiver side. There are two major concerns regarding the audio codec, namely resolution and sampling speed.

Regarding resolution, the current norm in audio codec supports 16 bits, which can represent 2^{16} = 65536 possible levels. Since the audio interface can be designed to amplify or attenuate input and output signals to or from the audio codec block, a 16bits audio codec can easily represent sinusoids with negligible quantization noise.

As for the sampling rate, the radio modules are capable of transmitting and receiving at 300Hz to 3kHz. In order to satisfy the Nyquist sampling rate, a minimum of 6kSa/sec must be satisfied. Therefore, an audio codec operating with a sampling rate of 8kSa/sec should be able to provide a reasonable representation of the original waveform without significant deteriorations.

1.4 Audio Interface

The main function of the audio interface is to amplify or to attenuate signals as required, so as to ensure signals can be transmitted or received clearly regardless the choice of radio modules. With tune able potentiometers, op amps can be used to amplify or attenuate signals from the audio codec to the radio modules as desired to ensure clear transmission. According to the project specifications, the audio interface should allow the audio input and output levels to be adjusted such that it falls between -20dBV and 0dBV. Since most op amps can deliver such performances, this requirement can be satisfied with the appropriate choice of op amps and potentiometers.

2. Radio Link Characterization

The voice-band radios used with the modem play a key factor in determining design decisions, particularly for the modulator and demodulator blocks. For this reason, it is necessary to have at least some metrics by which the radios can be characterized.

The Cobra MicroTalk PR-245-2 GMRS radio was chosen for use in this project due to its low cost and simple audio interface that will allow connection to the modem. This 1-watt UHF-FM radio provides a relatively clean voice-grade link spanning several kilometers. The modem will not be limited to functioning with this particular radio, but should function well with higher-performance, higher-power UHF voice-band radios.

Since the particular characteristics (such as usable audio bandwidth and noise floor) of this radio were unknown, several tests, two of which are detailed in the following discussion, were performed to provide performance characterization. It should be noted that all of these test were performed indoors with the radios 10 cm apart. Thus, they represent the best-case conditions, or an upper bound on voice-link performance. The PR-245-2 supports both FRS (Family Radio Service) and GMRS (General Mobile Radio Service) channels. The maximum allowable transmit power for GMRS channels is 1 watt as opposed to the 0.5 watt limit of FRS channels. Additionally, channel spacing is wider over GMRS channels. The tests discussed here were performed over FRS channels.

2.1 Frequency Response

A vital parameter in the design of the modulator and demodulator is the bandwidth available for transmitting data. Figure 2 illustrates the measured frequency response of the end-to-end bandpass voice link. A frequency swept tone was transmitted from one radio while the received amplitude response was measured at the output of the other radio. It should be noted that the vertical offset of the response (or the passband amplitude) is somewhat arbitrary since it depends upon the gain settings of the receiving audio equipment. What does matter is the relative variation of the response. From Figure 2 it should be noted that frequencies below 300 Hz and above 2000 Hz are significantly attenuated. Thus, as long as the modulated spectrum is constrained within the passband sufficiently the frequency response should not present an obstacle to data transmission at sufficiently low data rates.

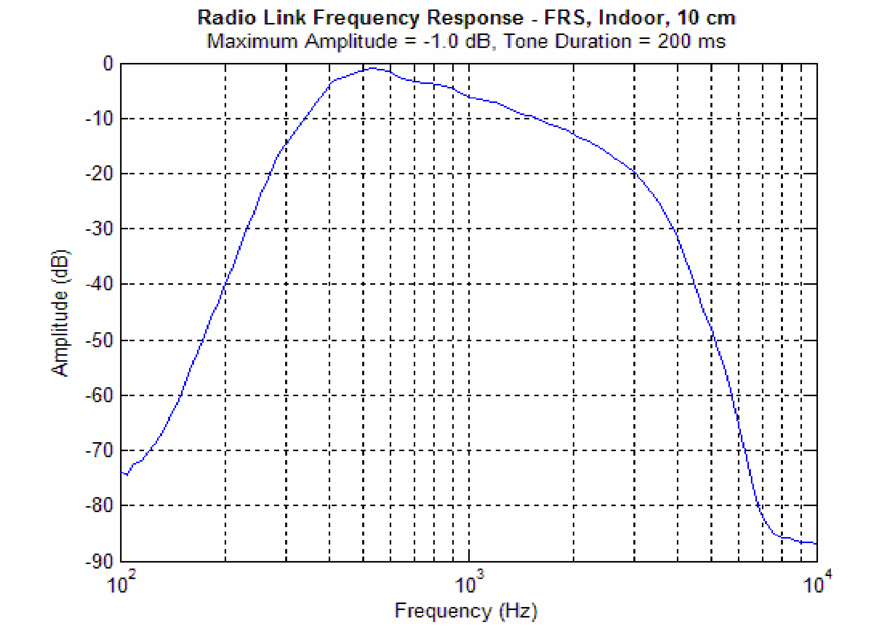

Figure 2 - Frequency response of voice-band channel

2.2 Noise Floor

Another useful metric is the noise floor of the radios at close range. Figure 3 illustrates the received power spectrum of a 1000 Hz test tone transmitted between the two radios. The amplitude of the transmitted tone was maximized while attempting to keep the power of the odd harmonics from increasing due to audio circuitry distortion in the radios.

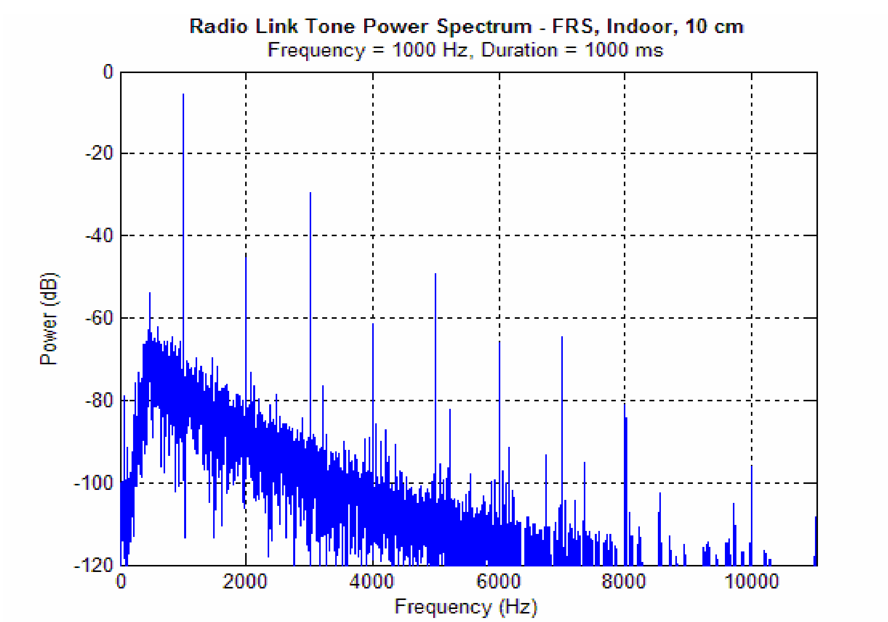

Figure 3 - Power spectrum of test tone

Several observations are now possible. This measurement represents a best-case channel condition: maximum transmitted signal power with minimum path loss. As path length is increased, the tone will eventually disappear into the noise floor. Also, interference from other sources will serve to increase the noise floor. Noise conditions over longer path lengths will not be explored here, but will be left to the bit-error-rate performance simulations and/or measurements in the final report.

The harmonic distortion introduced by the radios is also evident, particularly at the odd harmonics. Non-linear amplification and FM modulation imperfections contribute to this distortion.

The SNR (Signal-to-Noise Ratio) for the test tone under these conditions was measured by bandpass filtering the receiver output both in the presence and absence of the tone and comparing the relative measured power. The equivalent noise bandwidth (BW_N) of the filter and the SNR (C/N) of the tone were calculated to be:

$$BW_N = 1006.95\ \mathrm{Hz}$$

$$\frac{C}{N} = 10\log_{10}\left(\frac{P_{TONE}}{P_{NOISE}}\right) = 35.48\ \mathrm{dB}$$

From these figures, and with knowledge of the desired baud rate, it is possible to estimate the ratio of bit energy to noise power density for a particular data rate under ideal conditions.

3. DSP Software Architecture

3.1 Overview

The DSP software project is clearly divided into two parts, the lower-level firmware and the modem software and an optional debug component. The firmware is responsible for performing I/O operations and controller external devices. The debug component produces an command-line interface through the serial port, allowing the developer to configure modem options and debug the modem while its in use.

The modem software is platform independent, so it can be reused as new technology arises. Another advantage is that exactly the same code used on the DSP can be developed and simulated on the PC. A high level overview of the DSP software block architecture is shown in

Figure 4.

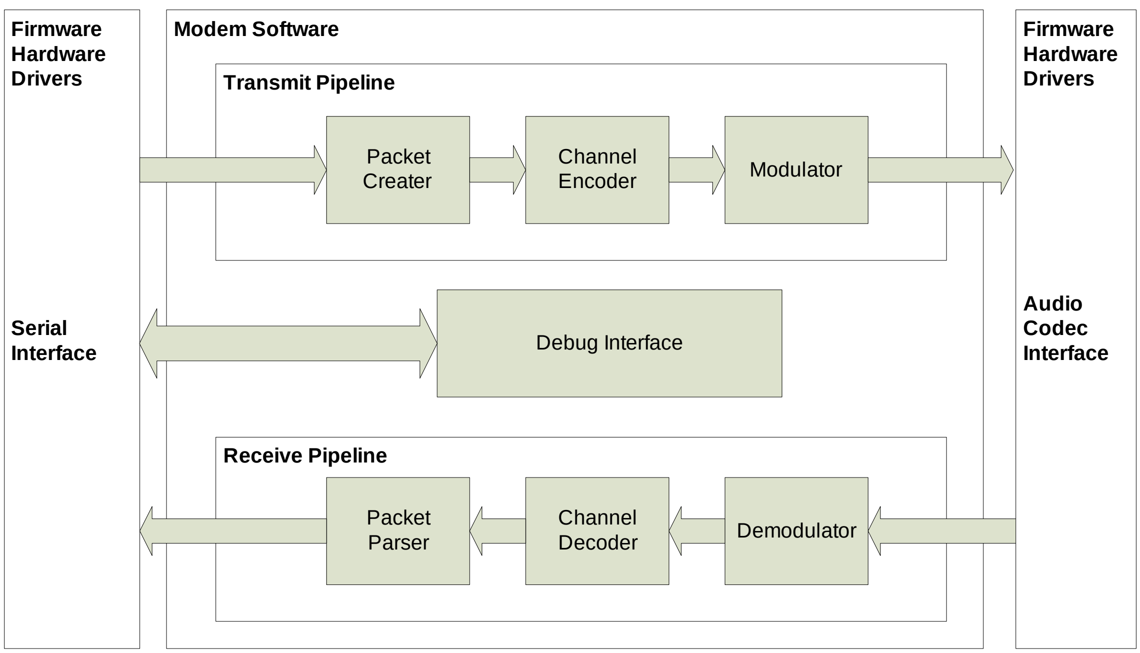

Figure 4 - Software Architecture Overview

The C language is chosen to implement the project. It was chosen for its wide acceptance and ease of learning. As opposed to higher level languages such as C++ and Java, C gives us more ability to optimize our software framework and its underlying pieces.

3.2 Software Methodology

Like all engineering projects, it is a good idea to split up a complex design into smaller chunks.

For the modem software, this methodology is enforced through the use of object-oriented design. The modem is organized into two types of objects, pipelines and controllers, using a customized object framework.

Each controller, or modem block, performs one processing stage of the modem, such as encoding or modulation. The controllers using a standard interface are then strung together into a single pipeline.

Another important advantage of the object-oriented design is that it allows multiple modems to be instantiated. This makes it possible to test interactions between two modems in the PC simulation environment.

3.3 Software Modem Blocks

3.3.1 Packet Encapsulation

The purpose of this block is to establish transmission synchronization and data flow control by adding header information into blocks of data. The main motivation of this block is to improve the reliability of the modem data transmission. Without this process there is no guarantee that all the packets sent to the receiver will be received. This is a problem for our application of the modem because every packet is crucial in our transmission session. Adding this packet encapsulation block will improve data error rate and accommodate interruptions in transmission by establishing “hand shaking” of the sender and the receiver.

The three major functions of this block are to achieve transmission synchronization, create packets on the sending end and parse packet information on the receiving end.

On the sending end, the block will encapsulate blocks of signal data and encapsulate them with meaningful information. Figure 5 contains a display of the contents of the packet header.

16	32	32	4	4	16	variable
sync bits	SEQ. #	ACK. #	Header length	code bits	check sum	DATA

Figure 5 - Contents of Packet Header

Sync bits are used to allow the system to match a unique sequence created by the computer to tell the receiver the beginning of meaningful information. The receiver will continue to parse information received until it finds the synch bits. If the received information is exhausted before the synch bits could be matched then the receiver will initiate a resend.

Source port and destination port numbers: They are used to give the system the capability to carry out several different transmissions simultaneously. Each unique transmission will have a different source and destination number to differentiate them from each other on the receiving end.

SEQ number and ACK number: Sequence number and acknowledge number is used to implement "hand shaking" between the sender and the receiver. Each octet of data will be given an unique sequence number. The receiver will send an acknowledgement number of the received sequence number plus one to the sender to notify the sender that it has successfully received the block of data. When the sender receives this acknowledgment it will send the next packet and this will continue until the end of the transmission. This is also called "handshaking".

Header length: The header length field contains information about the length of the header.

Code bits: This field contains the information about what type of packet this is. (Ie, SEQ, ACK, etc.)

Check sum: check sum is a mechanism used to ensure that there are no errors in the bits received. If the check sum and the sum of all the bits receive are not equal, we know that there has been errors in the received data.

Since the modem will not be used to stream high bandwidth, therefore, we have decided to set 100 bps as our minimum for our primary design objective. This goal was set based on the assumption that the modem's geographical operational areas will have low noise. We have also picked an simpler transmission scheme for our primary design objectives by sending and receiving one packet at a time. We have not decided the exact size of our data blocks that will be encapsulated in a packet because we need to still need to determine unknowns such as actual noise of the transmission media and time it takes for a sender to switch to a receiver.

Right now all we know is that there is a large physical/mechanical delay caused by the switch pushed to change the radio's from sender to receiver we have to assume that there will be low error rate in order to achieve our desired transmission speed.

This specific design of the packet encapsulation block is used based on research of current industry standard. This type of encapsulation scheme is proven to be effective in other modems. Our group has also decided to use a simple and achievable scheme to ensure that we would have a working product for our primary design objectives.

3.3.2 Soft-Decision Viterbi Convolutional Codec

The Convolutional codec is a forward error correction technique which is widely used in wireless digital communication systems where low SNR makes it highly susceptible to bit errors. Block codes, another forward error correction technique was also considered, but its ability to correct scattered bit errors is limited. There was also the possibility of concatenating block code with convolution code, which has proven performance advantages. However, this increases implementation complexity and performance requirements. For the same reason, turbo codes (the best performing FEC code) were not chosen.

The Convolutional decoder is implemented using a Viterbi algorithm, which has a fixed runtime and relatively reasonable processing requirements. It is important to note that the decoding algorithm only has an affect on the processing requirements and does not error-correction performance when identical parameters are chosen.

The BER (Bit Error Rate) performance of a convolutional encoder and decoder with known parameters, when subjected to Additive White Gaussian Noise is well defined and documented. These parameters are the output rate, code length, polynomial coefficients, and length of history bits kept inside the trellis. The codec is also soft-decision, which means that uncertainty levels of received binary signals are passed to the decoder. These parameters have to be set based on tests with real signals and adjusted for the processing power of the DSP used.

At this stage of modem development, only the PC simulation environment is available, and so a simulated AWGN channel was created to compare the performance of the designed Convolutional codec to existing referenced designs. The BER vs SNR performance is shown in Figure 6.

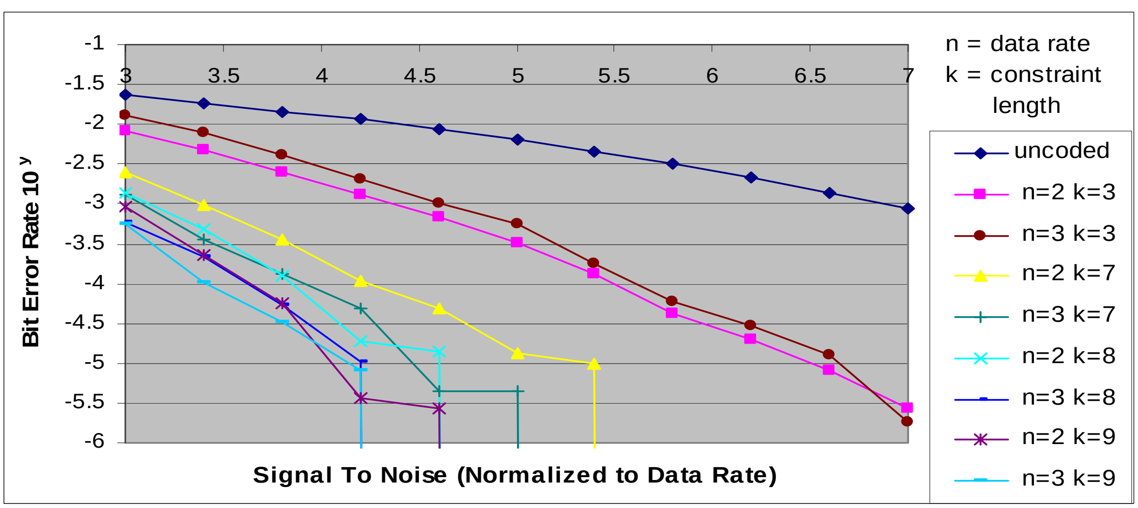

Figure 6 - BER vs. SNR for Soft-Decision Viterbi Convolutional Codec

The codec parameters are configurable at run-time, which allows for easy testing. The results are shown below. The BER (Bit Error Rate) vs. SNR graph is normalized for different data rates (which some books usually omit).

The n and k parameters, in the chart above, represent the output data rate and constraint length parameters respectively.

As the convolutional data rate is increased, there is very little effect on the BER performance. This is because the chart is normalized with respect to higher data rates, which offsets any error-correction benefits.

However, the BER performance increases significantly as the constraint length is increased. Larger constraints cost nothing in terms of bandwidth, but are very expensive in terms of processing power. In fact, there is an inverse exponential relationship has constraint lengths are increased, as shown in the graph below.

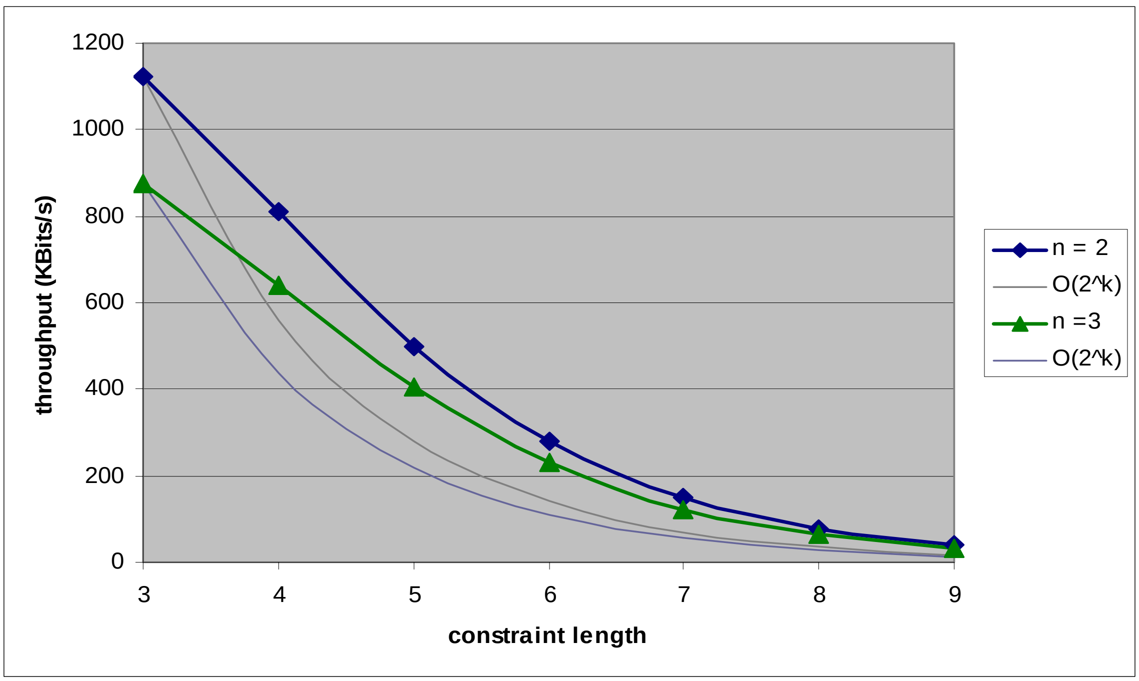

Figure 7 - Codec performance vs. Constraint Length

In conclusion, the ability of the convolutional codec to improve Bit Error Rates is well proven and confirmed with simulation. The next step in the design phase is to optimize the codec parameters for the intended application and to push constraint length as high as high as the processing resources allow.

3.3.3 Modulator and Demodulator

Several aspects that affect the modulation scheme design will now be considered. Modulation schemes that have constant or relatively constant envelopes are less sensitive to fluctuations in amplitude and non-linearity in amplification. Thus in applications where these effects are factors, angle modulation of some form is generally preferred. Two forms of angle modulation will be considered here: FSK (Frequency-Shift Keying) and PSK (Phase-Shift Keying).

When encoding the transmitted data the net data rate is scaled by the rate of the data code. This can be offset by choosing a multilevel modulation scheme that compensates for the loss in data rate by increasing the number of bits transmitted per symbol. In the case of a one-half rate convolutional code, the net data rate will remain unchanged if two bits are encoded in each transmitted symbol. For the cases of FSK and PSK this entails using 4-FSK or 4-PSK (also known as QPSK).

Figure 8 illustrates the uncoded error rate curves for these two modulation schemes. Instead of using SNR (Signal-to-Noise Ratio) as the horizontal variable, a normalized metric of bit energy per noise power density (E_b/N_o) is utilized. This is the ratio of the energy contained in one transmitted bit divided by the noise power per Hertz of bandwidth. Implementations with different symbol rates (baud rates) and different receiver filter bandwidths can be conveniently compared by converting SNR figures to $E_b/N_{o.}$

The upper curve represents the error rate for non-coherent, orthogonal 4-FSK. In this modulation scheme, four tones are used to encode all four possible values of a symbol of two bits. Non-coherent receivers do not rely on phase synchronization for the demodulation of the modulated signal. The advantage of this is simplicity of implementation. The cost is reduced performance over coherent demodulation schemes. Orthogonality implies that the various tone frequencies are spaced such that they do not interfere with one another when filtered ideally.

The lower curve represents the error rate for (coherent) differentially encoded 4-PSK. In this scheme, the four possible values of a two-bit symbol are encoded into four equally spaced carrier phases (each 90° apart). Coherent demodulation is used to recover these shifts in carrier phase. Additionally, the bits are usually differentially encoded so that the exact starting phase is not required to recover the data, only the relative phase shifts.

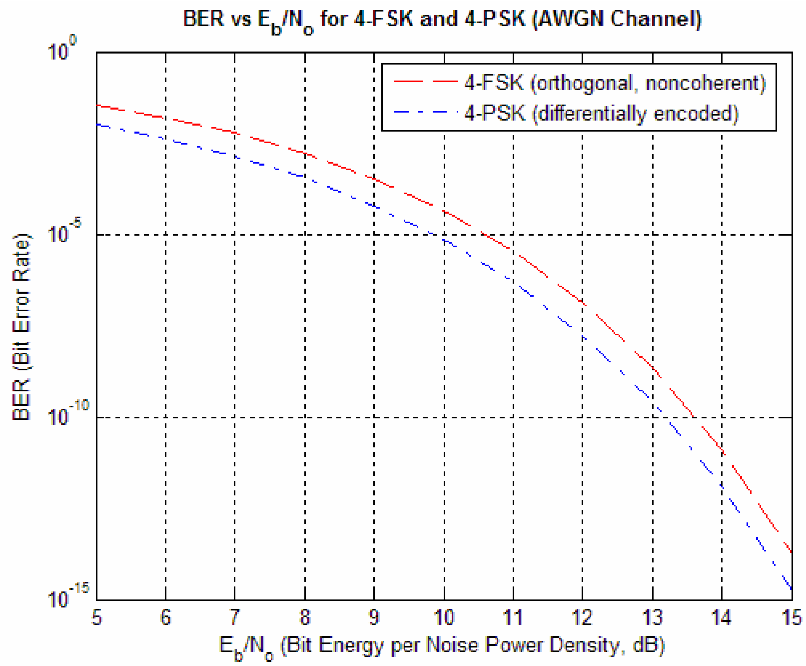

Figure 8 – Error rate curves for 4-FSK and 4-PSK

To evaluate the best-case theoretical performance of these two modulation schemes with various SNRs, it is necessary to convert SNR figures to E_b/N_o. This can be done using

$$\frac{E_b}{N_o} = \frac{1}{n_b} \cdot \frac{C}{N} \cdot \frac{BW_n}{f_b},$$

where C is the carrier power, N is the noise power, BW_n is the equivalent noise bandwidth of the receiver (or the noise-measuring filter), f_b is the symbol rate (or baud rate), and n_b is the number of bits per symbol.

With our previously measured noise floor for the radios used in this project, the following results were obtained:

$$\frac{C}{N} = 10^{\frac{SNR}{10}} = 10^{\frac{35.48}{10}} = 3531.8 \text{ and } BW_n = 1006.95\,\text{Hz}.$$

This yields

$$\frac{E_b}{N_o} = \frac{1}{n_b} \cdot \frac{C}{N} \cdot \frac{BW_n}{f_b} = \frac{3531.8(1006.95)}{2(100)} = 17781.89 = 42.5 \text{ dB}$$

for a symbol rate of $f_b = 100$, corresponding to the minimum data rate set in the project specification (100 bits per second) while allowing for a one-half rate code. From Figure 8, it can be seen that this best-case condition of E_b/N_o will result in an extremely low BER for both 4-FSK and 4-PSK without considering any coding improvement.

The analysis can also proceed in the reverse direction. In the project specification, the maximum acceptable BER for usability was set at 10^{-3}. The E_b/N_o corresponding to this condition (neglecting any improvements due to coding) can be estimated from Figure 8 to be near E_b/N_o = 8.3 dB for the case of 4-FSK. Working backwards, it can be seen that this requires a C/N ratio of

$$\frac{C}{N} = n_b \cdot \frac{E_b}{N_o} \cdot \frac{f_b}{BW_n} = 2 \cdot 10^{\frac{8.3}{10}} \cdot \frac{100}{1006.95} = 1.342 = 1.3\,\text{dB}.$$

The C/N radio for the same error rate using 4-PSK will be even lower. All of these numbers assume AWGN and an ideal demodulator. In reality, neither of these conditions will be achievable. Thus the numbers above serve only as bounds on the expected modem performance. Actual (simulated and/or measured) error rates for the implemented modem will appear in the final report.

One additional aspect that must be explored is the power spectrum produced by each modulation scheme. This is important since the voice-band radios provide a band-limited channel for transmission. Figure 9 illustrates a typical power spectrum of a modulated square wave for a single carrier frequency (PSK or on-off keying). While the spectrum theoretically extends indefinitely, it is convenient to approximate the minimum bandwidth required for transmission as the width of the central lobe, as it contains most of the signal power. Moreover, the height of the other side lobes can be reduced significantly with appropriate filtering.

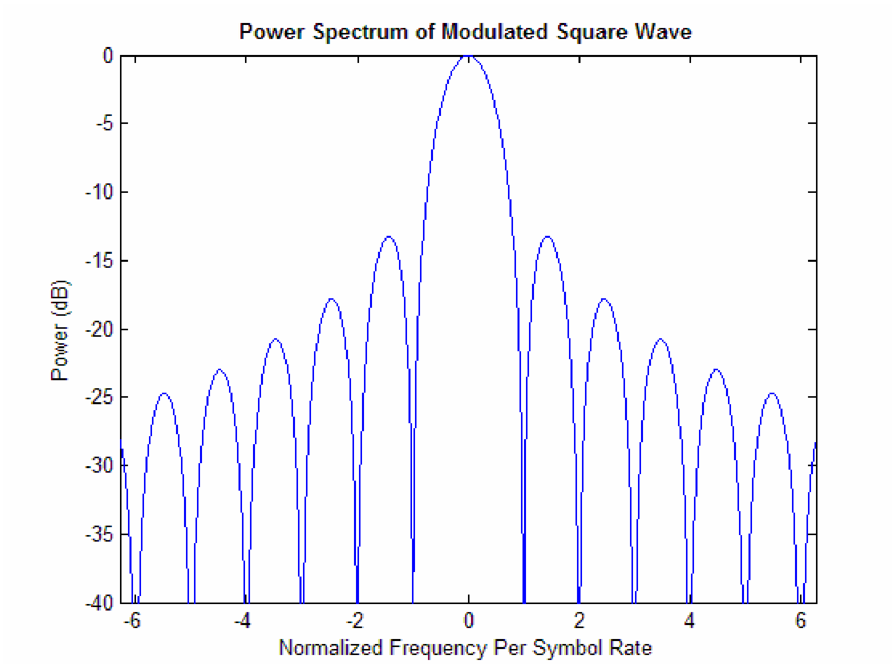

Figure 9 – Typical power spectrum of a modulated square wave

The width of the main lobe is

$$W = 2 f_b ,$$

where f_b is the symbol or baud rate of the baseband digital signal. This is seen in Figure 9 by the fact the main lobe extends from –1 to 1, representing a (normalized) frequency width of twice the symbol rate.

A 4-PSK modulated signal contains a single carrier with varying phase. Thus, the bandwidth required for a symbol rate of 100 symbols per second is estimated as the width of the main lobe:

$$BW = W = 2 f_b = 2(100) = 200 \text{ Hz} .$$

4-FSK modulation can be thought of as four independent carriers each modulated with on-off keying such that only one is on at any time instant (at least for the case of square-wave baseband data). For orthogonality in non-coherent demodulation it is required that a given carrier be located at the spectral nulls of all other carriers. From Figure 9 it is evident that the shortest possible spacing between carriers is thus one, or f_b, the symbol rate. In this case, the required bandwidth of 4-FSK for a symbol rate of 100 symbols per second becomes the width of all the main lobes when spaced appropriately:

$$BW = (M + 1)W = (M + 1)2 f_b = (4 + 1)(100) = 500 \text{ Hz} .$$

From the radio characterization data, it was observed that frequencies below 300 Hz and above 2000 Hz are significantly attenuated. Moreover, the response is flattest between 400-800 Hz. From the above calculations, it should be possible to work with either 4-FSK or 4-PSK within the bandwidth of the specified radios.

In summary, 4-PSK has advantages over 4-FSK in both error rate performance and required bandwidth. The downside to 4-PSK is the increased implementation complexity required by coherent demodulation to continually track the received carrier phase. 4-FSK offers good performance with the advantage of implementation simplicity for non-coherent demodulation.