Report for

ECE 4910

Senior Project Design

DATA INTEGRATION

IN

MULTICARRIER REFLECTOMETRY SENSORS

Prepared by

Afshin Edrissi

Date: Apr 7, 2006

ABSTRACT

Afshin Edrissi (Cynthia Furse), Department of Electrical and Computer Engineering,
University of Utah, Salt Lake City, UT 84112

In order to analyze faults on multiple branches in transmission lines and establish communication to the outside world using Multi-Carrier Reflectometry (MCR) sensors, it has become necessary to add data capabilities into the MCR sensors. Before hardware implementation in Very High Speed Integrated Circuits Hardware Description Language (VHDL), a series of simulations using Matlab, Simulink, and Modelsim was done in order to verify the validity of data transmission in the sensors. The design required a minimum of 24 real-time adjustable data channels with a minimum bandwidth of 2 kHz per channel. To facilitate meeting the design requirements, digital implementation was chosen, because there is already a Digital Signal Processing (DSP) chip associated with the MCR sensors. Thus with minimal changes to the existing hardware, new functionality can be added to sensors. In addition, digital processing allows for more precise control over the frequency band and signal processing, with easily changeable configuration for future implementation.

To modulate the data with a carrier, BPSK was chosen, since it provides a balance between ease of implementation and bandwidth efficiency. The simulations showed promising results for integration of data into MCR sensors at any desired frequency. The final implementation yielded 28 real-time adjustable channels with the frequency separation of 6.1 kHz starting at about 10MHz in the frequency spectrum. The start location was made variable in order to adjust the spectrum to application specific needs and to avoid as much interference as possible. The system has set the way for ease of implementation of algorithms that are more advanced and more robust data transmission.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

1.0 INTRODUCTION

Transmission lines are used everywhere, from powering everyday devices to transmitting power and data over long distances. This project focuses on the transmission lines that are used within the aircraft structure. A typical aircraft has many cables to provide power and data to different sections such as the lights, equipment, navigation, engines, etc. In order to increase the safety of the aircraft, it is very important to determine if the transmission lines, especially in critical systems, are in proper working condition. Multi Carrier Reflectometry (MCR) sensors are used to detect the faults on transmission lines. These faults can be open, short, or any impedance mismatch [1] due to other causes such as water dripping on the transmission line.

This project focuses on adding data transmission capabilities to these MCR sensors. There are many reasons why data communication is vital in the future of the MCR sensors. For example, while the aircraft is in the air, cold air can cause condensation of water vapor, which may cause wet areas where the water breaches cracks in the insulation of the cables, causing temporarily shorts or sparks between the lines. If these lines are not tested in real-time, it may be impossible to detect the faults after the plane has landed. It is important for these MCR sensors to operate continuously so if a fault is detected, it is able to communicate data to the outside environment such as the pilot or a technician on the ground. Yet another reason for data integration is to extend the capabilities of these MCR sensors to detect fault on branches of transmission lines (more than one line connected to a point) or to increase the accuracy of the fault location.

This paper will introduce the operation of MCR sensors as well as the methods used to integrate data into these sensors. Finally, some additional applications with the new capabilities of the sensors are discussed for future implementation.

2.0 METHODS

2.1 Operation and limitations of MCR Sensors

2.1.1 Operation

The concept of these MCR sensors is quite simple. A sine wave can be transmitted on the transmission line, where it will reflect back to the sensor, when it reaches the end of the line or an impedance mismatch such as a short circuit. Given all other conditions remain the same, the difference between the transmitted and the reflected waves is the phase shift caused by the delay of the reflected wave. This phase difference can be determined by taking the frequency spectrum of the transmitted and reflected signals using Fast Fourier Transform (FFT) and dividing them to extract the phase. Since the sine wave was locally generated and the frequency and the phase are known, the time delay (thus location of the fault) can be determined.

To increase the accuracy of the system, more than one sine wave can be used. The number of carriers is limited by the clock frequency and processing speed of the system. The current implementation of the MCR sensors is designed to use 64 carriers to determine the location of the fault. More information regarding these MCR sensors can be found in [1].

2.1.2 Limitations

The current design of the MCR sensors has a few limitations. They can only detect faults on a single transmission line. Another limitation is that there is a need for a processing device to analyze the reflection in real time rather than store the data and transmit it for further analysis if it detects changes in the reflected waves. By using data-integration, the capabilities of the sensors will expand in other ways to analyze and communicate fault location. This can provide more reliable sensors.

2.2 Data Integration

2.2.1 Design specification

Since the focus of the project is on the aircraft wiring, the design specification has been shaped by the physical properties of the aircraft wiring. The lengths of the wires are important because this limits the maximum detectable frequency. At this time, the sensors were designed to work with up to 100 ft of transmission line.

Another specification is imposed by the existing signals in the wires. The amplitude and frequency for these signals (1 Mbps for Data, 400 Hz for Power) [8] can be modeled as a Mil-Std 1553 signal (1 Mbps) **Error! Reference source not found.**. Given the specifications, the data can be modulated above the occupied bandwidth to avoid interference with existing signals on the line.

2.2.2 *Goals*

Beside need for data transmission, it is desired to have as much hardware as possible in Digital Signal Processing (DSP), since there is already a DSP system associated with the MCR Sensors. This can have numerous benefits such as smaller hardware, ease of transforming it to an Integrated Circuit (IC), lower power consumption, and higher accuracy.

In addition, it is desired to have multiple sensors communicate to each other simultaneously, where each sensor has its own communication channel. In this design, Frequency Division Multiplexing (FDM) is used in the data integration to provide 24 unique channels (frequencies) at any desired location in order to avoid interference with existing signals.

2.2.3 *Possible modulation methods*

One possible method of modulating data is to use the existing carriers that are used by the MCR sensors rather than to use a separate carrier at a different frequency. The problem with this method is that it will distort the frequency spectrum of the carriers and thus cause a problem for the sensor with fault detection. One way to try to overcome this problem is to use a very sharp band pass filter at both the transmitted and the received carrier and then take the output of the filters to the frequency spectrum for further processing [2]. This can be very costly in terms of hardware (such as 128 narrow-band Band Pass Filter) and software (high processing time), and decrease the accuracy the location of the fault.

Given the problems above, the project uses its own carriers to modulate and transmit data at a frequency that is not occupied by either the existing signals on the line or by the carriers that are used in MCR sensors. This in turn opens up possibilities of many different algorithm implementations such as FSK, BPSK (currently used), Spread Spectrum, Frequency Hopping, etc.

2.2.4 *Simulation and implementation*

Matlab^{tm} and Simulink^{tm} play an important part in designing and simulating this project, as many ideas needed to be tested individually and together to verify the functionality. Simulink, Xilinx ISE, System Generator, and Modelsim have been used to translate the models into hardware for further testing.

A Digital Signal Processing (DSP) board called *Xtreme DSP* has been used to implement the system in hardware. *Xtreme DSP* contains an Field-Programmable Gate Array (FPGA), high speed Analog Digital Converter (ADC) and Digital Analog Converter (DAC), Light Emitting Diodes (LEDs), and other analog components. In addition, the board is capable of using a parallel cable to communicate with the computer on a real time basis. Once the files that have been compiled using Xilinx's ISE, they can be uploaded directly to the board to start the hardware implementation.

2.2.5 *Transmitter*

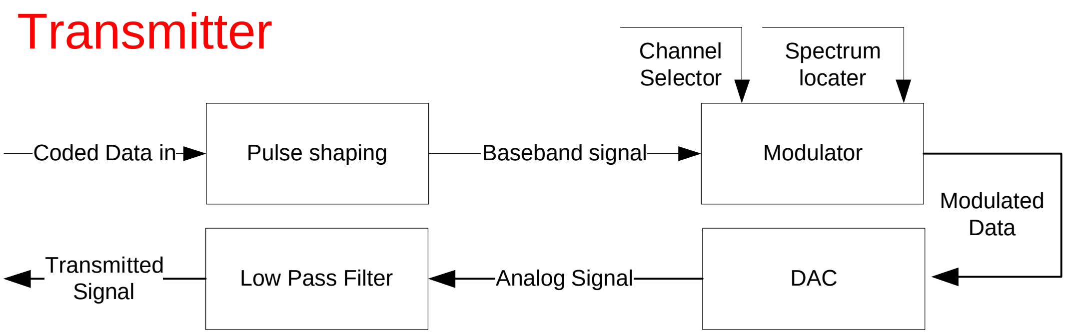

Figure 1 –Transmitter

The transmitter for this design consists of four stages as shown in Figure 1 –Transmitter, where the first three stages are done using digital signal processing and the last stage is done in analog circuitry to transmit the signal. Operation of each stage is described below. The transmitter expects Coded Data (any binary data that is understandable by the sensors) as its input and produces the analog modulated frequency ready to be transmitted using the transmission line.

Pulse shaping

Sudden changes to the transmitted signal will cause inefficiency in the frequency spectrum as well as Inter Symbol Interference (ISI)[4] if the analog pulse is wider than the time between adjacent symbols. In order to reduce the spectrum of the signal and decrease the ISI, a window for pulse shaping needs to be used. Given that it is more difficult, and higher processing time is needed to use a filter at high frequencies, the pulse shaping is done on the baseband signal, and then it will be modulated to shift its frequency spectrum.

A spectrum of a regular binary signal is shown Figure 2 – Pulse shaping as well as the same signal (smaller magnitude) passed through the pulse-shaping module. The spectrum passing through the pulse-shaping module is much smaller and decays more rapidly as the frequency increases. This can significantly decrease the frequency spectrum of the modulated signal, since the width of the modulated signal is twice the baseband signal.

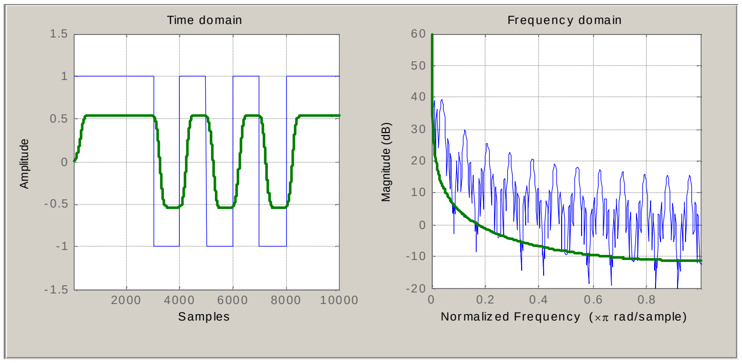

Figure 2 – Pulse shaping

Modulator

The modulator will shift the frequency spectrum of the baseband signal to any desired frequency. The heart of the modulator is Numerically Controlled Oscillator (NCO) and a fast multiplier. NCO can provide a sine wave of any frequency and phase digitally for the multiplier to multiply (modulate) the baseband signal with the carrier of desired frequency. The Figure 3 – Modulator shows the modulation process where the base band signal has been shifted in frequency spectrum to a set location.

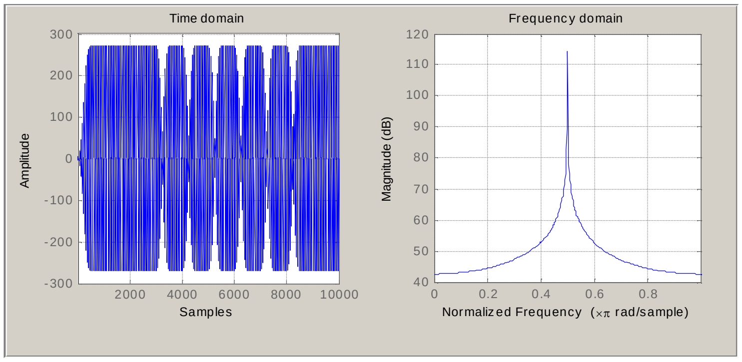

Figure 3 – Modulator

The operation of the NCO will be briefly discussed in the next section. Based on the Fourier properties, multiplying a signal by a cosine wave will shift the spectrum of the signal to the center frequency of the cosine wave[4]. This property allows translating the channel number to a specific frequency for the NCO and multiplying the output of the NCO with a baseband signal to generate the modulated output. At this stage, the phase of the carrier (from NCO) is not important, as the receiver will have a carrier recovery circuit to extract the exact phase and the frequency. The basic block diagram of the modulator where the carrier will be multiplied by the baseband signal is shown in Figure 4 – Modulator block diagram.

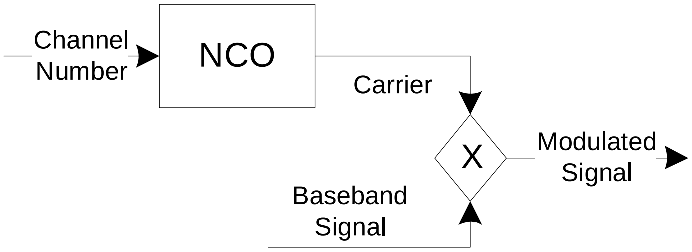

Figure 4 – Modulator block diagram

Once the baseband signal has been modulated, it can be added to the MCR waves (the sine waves that are used to detect the fault location) before converting to an analog signal for transmission.

NCO

The Numerically Controlled Oscillator (NCO) functions by storing the values of a cosine wave into the memory at the maximum resolution that is needed[5]. If the values (phase) are read sequentially, the resolution is maximized and the frequency is minimized. However, if at each time, every other value is read, then the frequency is twice the fundamental, and so on. Thus, the frequency corresponds directly to the number of values to skip at each time step. In addition to the values of the cosine wave, the values of the sine waves are also stored, so at any given phase, it is possible to generate a cosine and sine wave (90 degrees difference) at the same time.

Since the modulation method is BPSK, the resolution for channel separation corresponds to the resolution of NCO. Given that the clock of the FPGA is set at 100 MHz, and it is desired to have a minimum of 5 kHz frequency separation, the number of locations needed is $\frac{100\times10^6}{5\times10^3}=20,000$. However, since the system is digital, and the way the values for the phase are calculated is by accumulation and overflow[5], it is necessary to have a number of locations that is multiplied by two (binary number), thus $\lfloor \log_2(20,000) \rfloor = 14 bits$ are needed to represent $2^{14}=16,384$ locations given the resolution of $\frac{100\times10^6}{2^{14}}=6.104kHz$. This configuration gives a minimum frequency of about 6.1 kHz, and the harmonics of it. In addition, since the maximum possible frequency is half of the clock (50 MHZ), it is possible to generate $\frac{2^{14}}{2}=2^{13}=8192$ unique frequencies from $6.104kHz$ to $50MHz$ with the current configuration of NCO. These frequencies are grouped into the set of 24 adjacent channels called *Channel Number* for the sensors to communicate with each other and these groups are variable across the frequency spectrum using *Spectrum Locator*. This way it is possible use the occupied bandwidth of the data modulation a given application and avoid interference with existing signals on the line.

The values that enter the NCO are between 0 and 0.5 where it is possible to have a 2^{-14} resolution such as $\left(0\cdot 2^{-14}, 1\cdot 2^{-14}, 2\cdot 2^{-14}, 3\cdot 2^{-14}, \ldots, 0.5\right)$ with corresponding frequencies of $\left(0Hz, 6.1kHz, 12.2kHz, 18.3kHz, \ldots, 50MHz\right)$. The default *Spectrum Locator* is set to be centered at 10MHz, where it is roughly $\frac{10\times 10^{6}}{100\times 10^{6}} = 0.1$, or exactly the closest value that can be represented by 14 binary points ($0.099975586\ldots$). By manipulating the *Channel Number* and *Spectrum Locator* using a simple math, it is possible to adjusts the desired frequency spectrum.

A phase change can be represented by changing the time of a wave. Thus adjusting a phase of the NCO is possible with reading a different memory location, which has a constant offset. Once the NCO determines what memory location should be read, a number can be added (or subtracted) to the location of the memory to read a different location by an offset. Since a block of memory in NCO is read constantly and sequentially, this will change the time of the wave and thus the phase of the generated output.

DAC

The Digital to Analog converter takes the numbers that are generated from the modulator block and generates an analog voltage that can be transmitted. Since all the numbers generated are from -1 to +1, the DAC is setup accordingly to get the highest resolution possible. DAC of the Xtreme DSP can operate at 14-bits up to 160MSPS[7]. This is much higher than the maximum frequencies the system can generate due to its internal clock thus the DAC has been setup to generate 100 MSPS at 8 bits resolution (255 possible analog outputs).

Low pass filter

The purpose of the low pass filter is to remove the rough edges that are associated with converting a digital signal into an analog signal as shown in Figure 5 – DAC and low pass filter, where the signal with the rough edges are the output of the DAC, and the smooth signal is after being passed through a low pass filter.

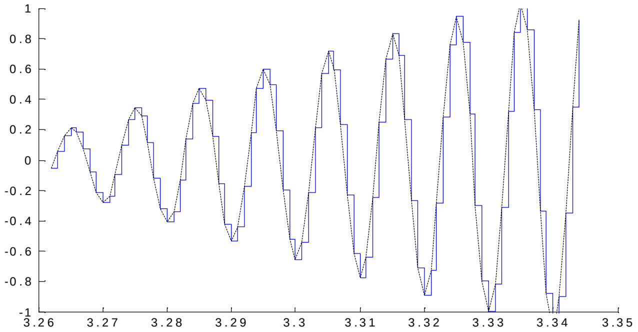

Figure 5 – DAC and low pass filter

The design of the filter was done using Matlab's FDAtool in signal processing toolbox to design the response and filter order graphically, and choosing the option *Realize Model* to get the hardware realization. This way, a time consuming task of filter design can be done quickly and accurately with capability of simulation in the Simulink environment. Although in this case any simple low pass filter would satisfy the conditions, in this design first order Butterworth was used with a corner frequency a little higher than the transmission frequency.

2.2.6 Receiver

The receiver in this design consists of six stages where the first two stages are in analog and the rest are done in the digital domain. Although one of the goals of this project is to design as much as possible in the digital domain, it is not possible to get away completely from the analog circuitry.

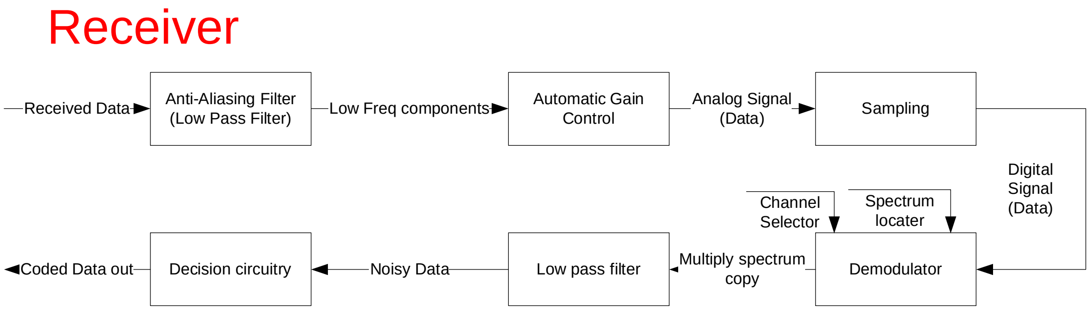

Figure 6 – Receiver Block Diagram

Anti-Aliasing Filter (Low pass filter)

Before the received signal can be sampled and taken to DSP for further processing, it is necessary to make sure that it does not contain any frequency components above the half of the maximum sampling rate (Nyquist criteria)[4] as explained in the *Sampling* section later. For the application of aircraft wiring, it is not a concern since all of the data and power 2.2.1 are well below the maximum frequency of 50MHz (100 MHz sampling). However, if the sensors are to be used in other applications with much higher frequencies, it is necessary to eliminate the high frequencies that are not associated with the data communication or they can interfere with the data. This can be easily done using a low pass filter with a corner frequency of about half of the sampling rate.

Automatic Gain Control

Since the sampling is done at finite resolution, it is desired to have the maximum amplitude of the signal correspond to the maximum sampling point of the Analog to Digital Converter (ADC). This way, the full resolution of the ADC can be used, which can distinguish smaller parts of the received signal. This is achieved in DSP by finding the maximum of the received amplitude with a very long time period (in seconds) thus adjusting the gain until the maximum reaches approximately the maximum value that can be represented by the ADC[4].

Sampling

Sampling is an integral part of any digital processing system. It provides a way for the DSP system to communicate to the outside (analog) world. The faster the sampling rate of the system the higher frequencies it can process. Due to the Nyquist Criteria, the maximum frequencies should be no more than half of the sampling rate. The capabilities of the Xtreme DSP board are 14-bit and up to 105 Mega Samples Per Second (MSPS)[7]. In this project, the system is set to 8-bits at 100 MSPS, since at this rate frequencies up to 50 MHz can be processed. Once the incoming analog signal has been converted to digital, the DSP can begin to operate on the numbers to extract the *Coded Data* that was transmitted from it.

Demodulator

The demodulator as seen in Figure 7 – Demodulator, time and frequency can shift the frequency of the modulated signal back to its original spectrum. However, in this process another replica of the frequency spectrum will be generated at twice the carrier frequency, which will be removed later using a low pass filter.

It is very important to demodulate the signal with the exact frequency and phase of the carrier[6]. Given that the transmitter and receiver are crystal controlled and of the same type of device, this project focuses on estimating the phase of the carrier and assuming that the frequency is exact.

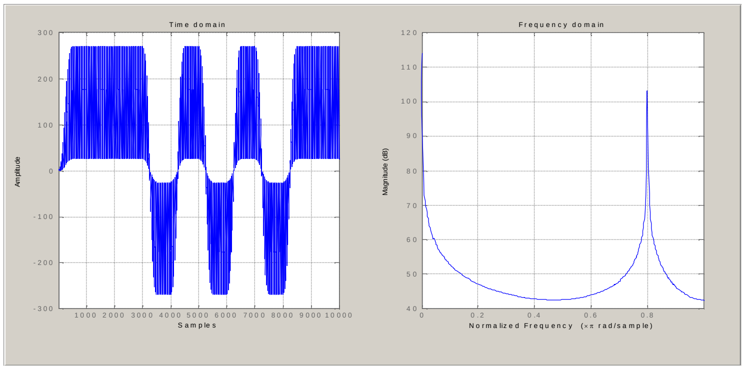

Figure 7 – Demodulator, time and frequency

The phase is difficult to determine, since the signal will travel an unknown distance from the transmitter to the receiver. Thus, the larger the distance, the longer time it takes for the signal to be received. This time corresponds to phase shifts between the signals. Even if the distance was known exactly, it is very difficult to determine the time when the oscillator for the transmitter was turned on. All this and more demands a carrier recovery module. After the phase from the carrier recovery module has been determined, then it is possible to use the information to demodulate (multiply) the incoming signal.

Carrier Recovery

As mentioned above, Carrier Recovery plays an important role in the demodulation process. A modified Phase Locked Loop (PLL) is used in this project for carrier recovery as shown in the Figure 8 – Carrier Recovery.

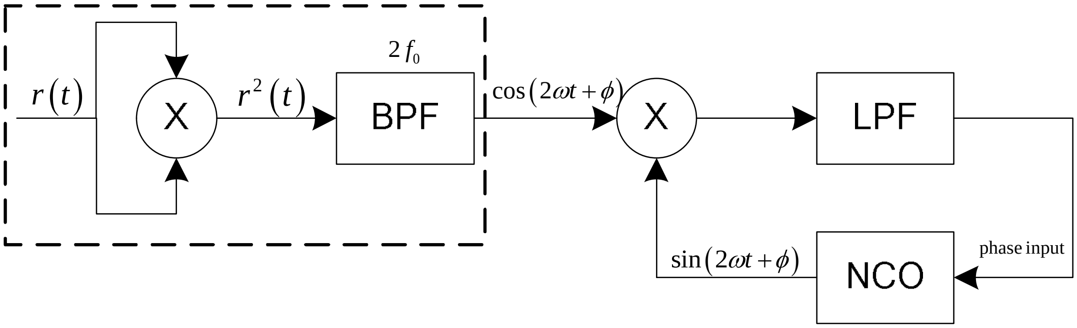

Figure 8 – Carrier Recovery

The box around the first part of the Carrier Recovery module is a standard pre-processing that is done to the received signal in order to extract the carrier [6]. Next, the carrier will be demodulated to DC with the goal of reaching zero. The only way to have a zero DC after low pass filtering is to demodulate the carrier with another carrier of the same frequency that is 90 degrees out of phase. Some simple mathematical operations are done to the output of the LPF to feed it directly to the input of the phase difference of the NCO. This will cause the phase to change if there is a non-zero input. The phase of the output of the NCO will change until a zero is presented at its input, and at this point it will *lock* to the 90 degree offset of the input. As mentioned before, The NCO has two outputs with 90 degree phase differences. If the sine output is used in the carrier recovery, at the same time the cosine output will provide the carrier to demodulate the received signal [6].

Low pass filter

As seen in the previous step, through demodulating another copy of the spectrum at a frequency centered at twice the carrier frequency will be created. Since the frequency is much higher than the baseband used by the signal, it can be removed by a low order low pass filter. As before, the filter was designed and realized by Matlab's function *FDAtool*. Once the signal has been passed through the low pass filter, the entire high frequency component will be eliminated, and thus the baseband signal remains as shown in the Figure 9 – Low pass filter, time and frequency.

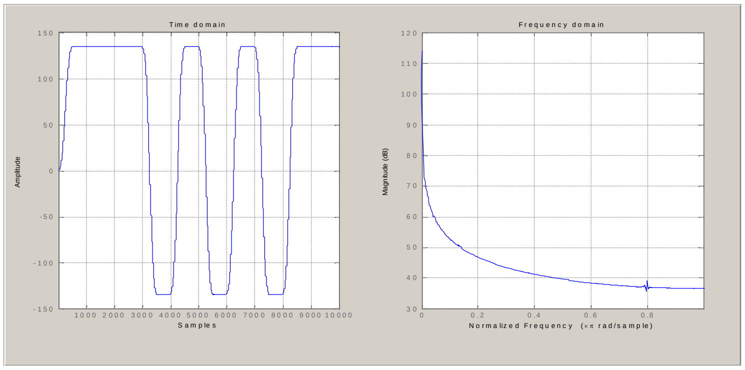

Figure 9 – Low pass filter, time and frequency

Decision Circuitry

At this point, the output of the low pass filter should resemble the transmitted data. However, this output can contain small noise. A decision circuitry is used which will compare the values (range from -1 to 1) to zero and based on that result, it will generate a clean digital output. The output of this stage should be identical to the input of the transmitter, thus successfully transmitting data to another location.

3.0 RESULTS AND DISCUSSION

As shown previously, it is possible to transmit data at any desired frequency between the transmitter and the receiver. With the current implementation, the incoming data is sampled at a fixed rate of 1 kilo samples per second and is modulated, transmitted, and the receiver will output the same data rate.

One of the goals of the project was to design 24 separate channels so different sensors can communicate simultaneously without interfering. With the current design, it was possible to have up to 28 channels without change of hardware (due to sampling, filters, etc.) which overcomes the minimum required number of channels.

In addition, as mentioned previously, the minimum desired separation between the adjacent channels were 5 kHz. With the limitations of the NCO as previously discussed, each channel will be separated by about 6.1 kHz. This provides enough guard bands so the adjacent channels do not interfere with one another.

Since having a flexible location for the channels in the frequency spectrum is desired, the design incorporates it into the system. This way, the transmitter and receiver need two inputs, the *channel number* and the *spectrum locator*, where the spectrum locator can change the frequency on larger scale for the entire channels group.

4.0 CONCLUSION AND RECOMMENDATIONS

4.1 Future implementations

This project has set the foundation for future implementations of new algorithms and designs. This is an open-ended design where many new changes can be easily programmed into the system and deployed to the consumers without additional changes to the hardware implementation. In addition, other methods such as spread spectrum can be used to decrease the amplitude of the data transmission in order to cause much less interference with its environment.

4.2 Applications

The application for the MCR sensors does not end with aircraft wiring. The number of applications are endless. For example, in a car, there are systems that transmit data or power to critical locations and it is important to know the conditions of the transmission line to increase the safety and to decrease the down time. Power grids can be another example where the system can be used in real-time to know the status of the wires and in case of a fault, find the location of the fault fast and minimize the hazard and downtime. These are not the only applications for the sensors; the system can be expanded to any data or power lines for any devices or environments that use wires!

REFERENCES

[1] Suketu Naik, Cynthia M. Furse and Behrouz Farhang-Boroujeny, "Multicarrier Reflectometry," accepted to IEEE Journal of Sensors

[2] "MIL-STD-1553 DESIGNER'S GUIDE SIXTH EDITION", http://www.ddc-web.com

[3] Dr. Farhang, "Transfer function identification Narrow band filters", IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 44, NO. 6, JUNE 1996

[4] C. Richard Johnson, *Telecommunication breakdown* Chapters 4, 6, 11. Prentice Hall, August 22, 2003

[5] Xilinx's technical note, "DDS", April 28, 2005, http://xilinx.com

[6] Swaroop Appadwedula, "Digital receiver: carrier recovery", July 2003, http://cnx.rice.edu/content/m10478/latest/

[7] Xilinx, "FPGA VIRTEX Data sheet", http://xilinx.com

[8] Dr. C. Furse, personal communication