2006-1029: A DIGITAL LOGIC BASED EXPERIMENTAL DESIGN OF A DSP/COMMUNICATION SYSTEM FOR ECET STUDENTS

Robert Adams, Western Carolina University

Robert Adams is an Assistant Professor of Electrical and Computer Engineering Technology in the School of Technology at Western Carolina University. His research interests include mathematical modeling of electrocardiographic applications, 3D modeling, and digital signal processing. Dr. Adams is a senior member of IEEE and a member of ASEE.

James Zhang, Western Carolina University

James Z. Zhang is an Assistant Professor of Electrical Engineering in the Kimmel School of Construction Management, Engineering and Technology at Western Carolina University. Dr. Zhang's research interests include Communications Theory, Wireless Networks, Bandwidth Efficient Modulation Schemes, Signal Design and Information Coding, and Digital Signal Processing Techniques for Communications. Dr. Zhang is a senior member of IEEE and a member of ASEE.

Paul Yanik, Western Carolina University

Paul Yanik is a Visiting Assistant Professor in the School of Technology at Western Carolina University and teaches in the Electrical and Computer Engineering Technology Program. His research interests include networking theory, electronics design automation and large system-on-chip verification. Prof. Yanik is a senior member of IEEE.

Kenneth Burbank, Western Carolina University

Ken Burbank is an Associate Professor and Director of Electrical Engineering at Western Carolina University. Dr. Burbank is active with IEEE, SME, and TAC of ABET, and strives to bring practical engineering activities into the classroom. His current project is the development of a photonics program within the Electrical Engineering curriculum.

A Digital Logic Based Experimental Design of a DSP/Communication System for ECET Students

Abstract

This article discusses the results of a senior design project to design a baseband system for a communication system, as well as on-going efforts to improve the design. Upon completion, the experimental design is intended for use in our Digital Circuits, Digital Signal Processing (DSP) , and Electronic Communications courses. For this reason, plans for incorporating design aspects of a communications system in these courses are discussed.

Introduction

In the past few years, the authors reported their efforts of enhancing students' learning by utilizing a systems approach[1,2,3,4,5]. These methods focus on the functionality of system blocks to improve students' understanding of system performance parameters. Positive results have been observed in strengthening students' vertical knowledge development on certain subjects.

However, weaknesses have also been found associated with the systems model. Most students can well understand the functionality of a system block, but they do not know how to realize the functionality by applying knowledge gained from other pertinent courses. For example, students may understand that a message generator generates appropriate digital streams for BPSK and QPSK modulations, but they may not be able to design this generator using digital logic circuits. This is due to their lack of experience in employing knowledge gained to the design of applications. To address this issue, we initiated a senior design project in the Spring semester of 2005, to design a digital logic-based DSP/baseband communication system.

This article reports some results of this senior design project, as well as our on-going efforts of refining this design. Upon completion, the experimental design is intended to be used in our Digital Circuits, Digital Signal Processing (DSP), and Electronic Communications courses. The following elements are incorporated into the design:

1) System clock design - A ring oscillator is designed to generate a 512 kHz system clock.
2) Analog-to-Digital Converter - An ADC0804 IC is used to convert an analog input into an 8-bit parallel digital output.

3) Sampling Circuit - A counter circuit is designed to sample the digital signal at pre-defined rates.
4) Parallel-to-Serial Data Conversion - A shift register circuit is designed to convert parallel data to a single data stream.
5) Data Stream Demultiplexing - A demultiplexing circuit is designed to allow user-selection of either a single data stream or two data streams of odd and even bits.
6) Phase Lock Loop - A PLL is employed to synchronize the received data stream(s).
7) Signal Combiner - A multiplexer circuit is used to combine the data streams to reconstruct the original digital information.
8) Serial-to-Parallel Data Conversion - A shift register circuit is designed to convert the serial data to 8 parallel data lines.
9) Digital-to-Analog Converter - An AD558 IC is used to convert 8-bit parallel digital data into the original analog signal.

The circuit boards designed in our senior project for data transmission and reception are shown in Figures 1 and 2, respectively. Further work is being done to refine our current design.

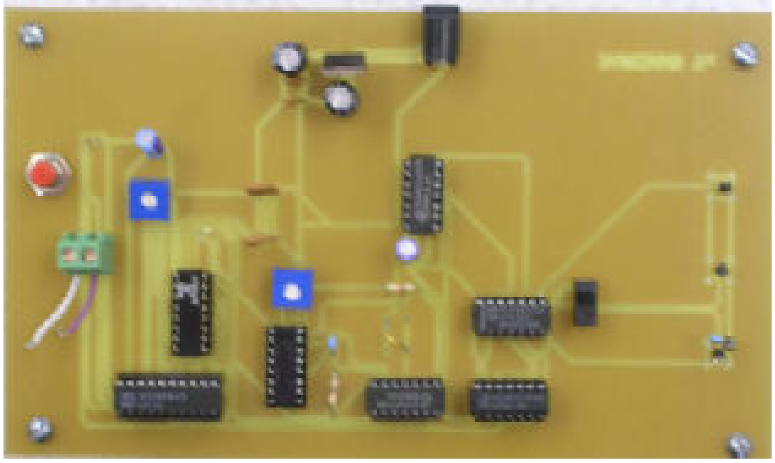


Fig. 1. ADC and Data Streaming Circuit Board

The goal of this project is to extend the students' horizontal knowledge base by introducing them to a tangible experimental design using knowledge gained from different courses. Successful implementation of this project will certainly help the students achieve this goal.

Digital Logic Design Process

Design Overview The baseband portion of the project contains the data transmission and reception function blocks shown in Figures 3 and 4, respectively. The transmitter section digitizes an analog input signal into 8-bit parallel values. For BPSK modulation, these values

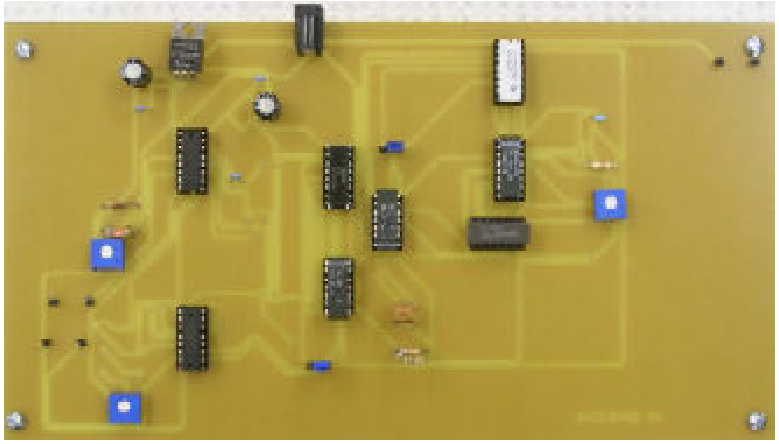


Fig. 2. Data Recovery Circuit Board

are multiplexed into a single output stream. For QPSK modulation, the values are multiplexed into parallel I and Q output streams of even and odd bits.

The receiver section performs the reverse operation. A phase locked loop extracts a clock from the incoming digital bit stream(s). For BPSK, a single incoming bit stream is converted to 8-bit parallel values. For QPSK, two bit streams are first recombined into a single stream and then parallelized. The parallel values are finally converted to an analog output, reconstructing the original transmitted input signal.

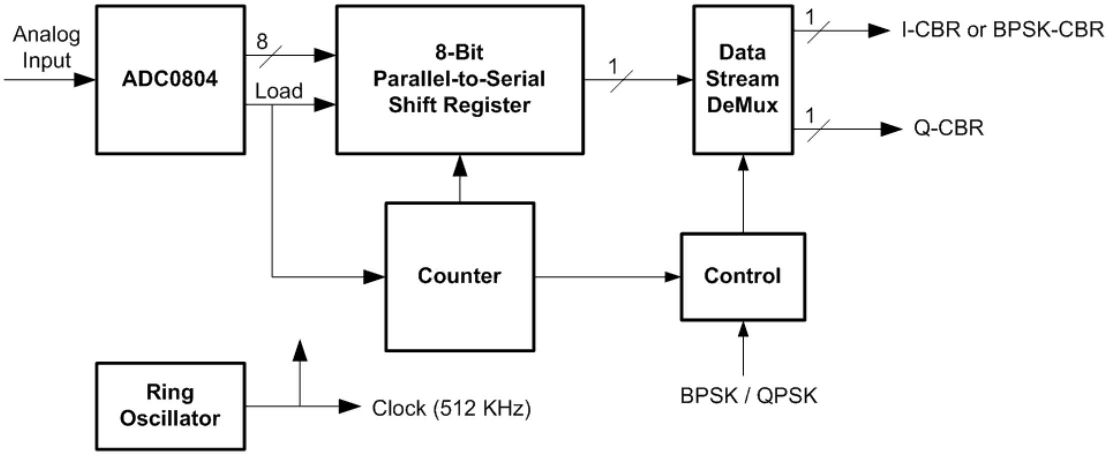


Fig. 3. Baseband Transmitter Block Diagram

Theory of Operation In the transmitter section, a 512 MHz clock is generated with a ring oscillator composed of three inverters. A variable RC network is used to set the desired frequency of oscillation. The generated clock synchronizes the components of the baseband transmitter.

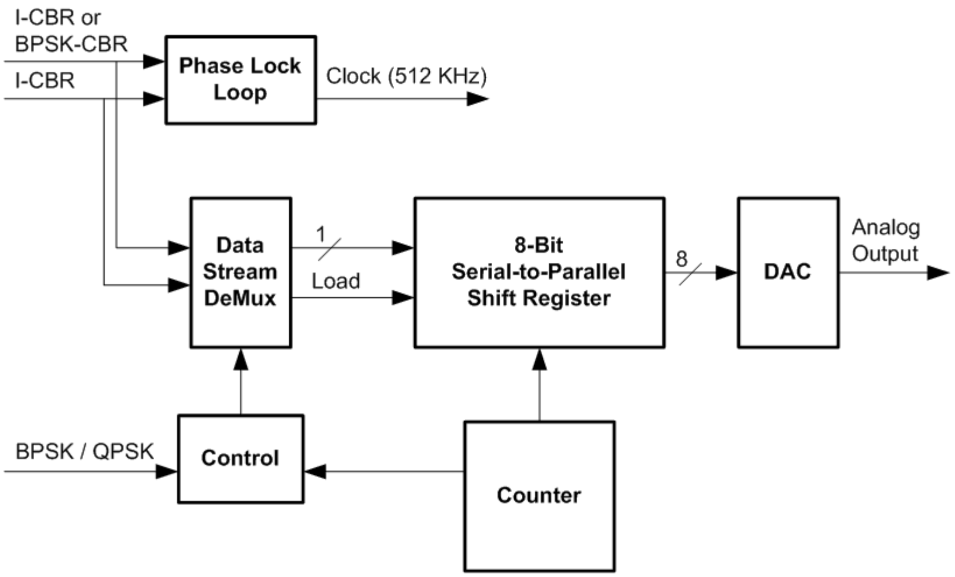


Fig. 4. Baseband Receiver Block Diagram

An analog input signal is digitized into an 8-bit parallel value using an ADC0804 A/D converter. The CS and WR signals from the A/D converter are logically combined to indicate when a new 8-bit value is available. As digital values are generated, they are loaded into a parallel-to-serial shift register. A counter is synchronized by the load indication from the A/D converter and controls shifting of the values out to form a serial bit stream. A control input selects either BPSK or QPSK modulation for the baseband digital output. For BPSK modulation, the serial stream is forwarded intact at 512 Kbps. For QPSK modulation, even and odd bits are demultiplexed from the serial bit stream to form two orthogonal serial output streams (I and Q) at 256 Kbps each.

At the receiver section, a phase lock loop extracts a 512 KHz clock from incoming digital serial digital bit stream. A control input indicates whether BPSK or QPSK modulation was used in transmission. For BPSK, the incoming digital bit stream is forwarded to an 8-bit serial-to-parallel shift register. For QPSK, I and Q bit streams are combined into a single stream with even bits extracted from the I stream and odd bits extracted from the Q stream. As with BPSK, the stream is forwarded to the shift register for conversion to 8-bit parallel values. A counter is used to delineate the 8-bit values. These values are finally passed to a D/A converter which reproduces the original analog signal.

Possible Improvements / Future Work While this design can be implemented using discrete logic and function-specific ICs, much of the design could be captured using a hardware description language (HDL). This approach would allow the student to confront and iron out design problems through simulation. While individual functions may present nominal design challenges, the interplay of design blocks facilitates a greater understanding of the design process and the underlying concepts of the communication link. A modest CPLD could be used for final design implementation. The debug process could be further enhanced at the prototype stage with extra pins of the programmable device being used to probe internal nodes.

Future work on this project might include the addition of features to accommodate higher-layer communication concepts. The addition of an encoding technique such as 4b5b to facilitate synchronization and link reliability would be a natural progression. Still higher functions might include the addition of a packetizing protocol and a form of forward error correction depending on predicted signal-to-noise ratios.

Application in a DSP Course

The hardware developed in our senior project can be used to address signal processing aspects of a communications system in a digital signal processing course. The fundamentals of sampling, A/D and D/A conversion, as well as software generation of a BPSK or QPSK signal are applicable topics to address in a DSP course.

Analog-to-digital conversion can be presented to students in terms of a three-step process: sample-and-hold, quantization, and binary encoding. This process is illustrated using the example shown in Figure 5 of a 3-bit A/D having a sampling rate of 1 sample per second and an encoder resolution of 3 bits per sample.

Digital-to-analog conversion can be presented to students in terms of a two-step process: a binary decoder followed by interpolation. The binary decoder processes bits in parallel and outputs a signal level in accordance with the decoding scheme. The output of the decoder is a discrete sequence $y[n]$. The interpolator multiplies $y[n]$ by a train of interpolation pulses to produce the output signal

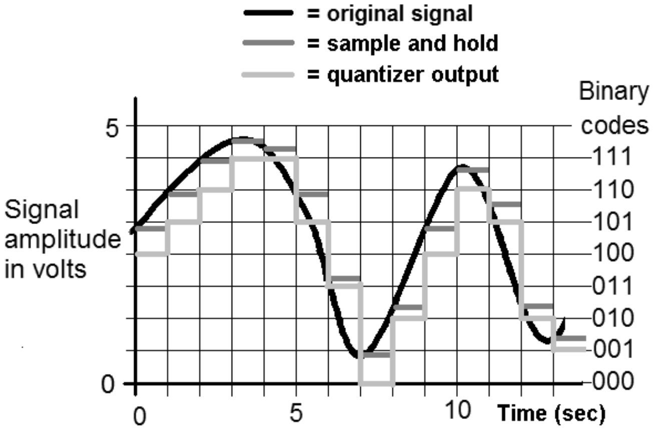
$$y(t) = \sum_{n=-\infty}^{\infty} y[n]p(t - nT_s) \qquad (1)$$

where $p(t)$ is the sampling pulse of the interpolator. The sampling pulse may be shaped to produce the smoothing effect of a low pass filter. Some examples of interpolation pulses include the square, triangular, parabolic, spline, and sinc pulses. It can be demonstrated mathematically to the students that the optimal interpolation pulse is the sinc pulse.

Students can use the hardware developed in our senior project for the purpose of understanding A/D and D/A conversion. Students can provide a DC input source to allow calibration of the ADC within a desired range of input signal and to correlate the input signal with the generated 8-bits of the ADC. Students can correlate the resulting 8-bits with the generated DC voltage of the DAC. Students can also use a function generator to provide a sinusoidal source to the ADC and a digital oscilloscope to simultaneously observe all resulting 8 output channels of the ADC, as well as the resulting sine wave generated by the DAC. Students can adjust the amplitude and offset of the sine wave so that at the the sine wave's minimum point, all ADC bits are zero, and at the maximum point, all ADC bits are one. This experiment will also enable students to observe the quantization and sample-and-hold properties of the ADC as well as the interpolation properties of the DAC.

A digital signal processor can be used to generate BPSK and QPSK. The TMS320C6416 DSP Starter Kit (DSK) combined with the Matlab® Link for Code Composer is a good combination of tools for allowing students to implement and demonstrate DSP processes. The Matlab® Link for Code Composer converts Matlab® code to a C code that can be ported through a USB link to the C6416 DSK for real-time processing. The C6416 has a Texas Instruments C6416 fixed point DSP along with A/D and D/A converters capable of providing sampling rates as high as 96 ksamples per second.

Students of a DSP class can write Matlab® code to generate the output of a BPSK or QPSK modulator. A digital stream $x[n]$ can be a repeated loop of a digital sequence stored in DSP memory. In order to generate a digitally sampled BPSK stream, or I and Q QPSK streams, the sampling rate must be chosen to be at least twice the carrier frequency. In addition the number of cycles per bit must be selected in the modulation sheme. As an example, if we were to sample at a rate of 96 ksamples/second with 8 samples per cycle, the resulting carrier frequency would be 12 ksamples per second. In addition, if 4 cycles per bit were to be generated, there would be 32 samples generated for each bit of $x[n]$. For a BPSK modulator, this would result in a data rate 96/32 = 3 kbps. For a QPSK modulator, this would result in I and Q data streams each flowing at rate of 1.5 kbps each. On the demodulator side, the digital samples of the BPSK modulator would be processed at 96 ksamples per second and fed into a decision process of estimating the transmitted bits. Groups of 32 samples from the BPSK modulator would be processed to make the decision for one bit of the received sequence $y[n]$. The received bits of $y[n]$ would be sent to the D/A converter on the DSK board at a sampling rate of 96 ksamples per second. The analog output of the DSK will be a data stream at a rate of 3 kbps. For the QPSK demodulator, the digital samples of the I and Q data stream would be each be processed at 48 ksamples per second and fed into a decision process of estimating the transmitted bits from each data stream. Groups of 32 samples from the each data stream would be processed to make the decision for one bit. The received bits from each data stream would be interleaved into a single data stream at a rate of 3 kbps which would be sent to the D/A converter on the DSK board. The accuracy of the output stream can be monitored on an oscilloscope. In addition, Matlab® code can be written by the student to generate and add noise to the digital samples of the BPSK or QPSK modulator. In this case, the bit error rate can be monitored by the DSP. A USB link between the DSK board and the PC allows the real-time display of DSP computations. For this reason a real-time monitor of the bit error rate can be displayed on the PC. The student would write the code to compute the bit error rate, but would not have to be concerned with establishing the PC to USB interface, as this has been provided by the DSK software and Matlab® has integrated the capability of displaying DSP omputation into a Simulink® block. In addition, DIP switches on the DSK board can be engaged to introduce different levels of noise. The student would have to write a Matlab® program to generate different levels of noise, but would not have to be concerned with the coding of interrupt hardware. Matlab® has integrated the interrupt handling capability into a Simulink® block of the DIP switches.



(a)

Time (sec)	0	1	2	3	4	5	6	7	8	9	10
Output bits	100	101	110	111	111	101	011	000	010	100	110

(b)

Fig. 5. A/D conversion example using a sampling rate of 1 sample per second and an encoding resolution of 3 bits per sample. (a) Graph of the original signal, sample-and-hold output, and quantizer output. (b) Resulting output bits generated by the binary encoder as a function of time.

Application in a Communications Course

The hardware described in the previous sections can also be used as an experimental apparatus for an electronic communications course. Because the device digitizes the analog input to generate a digital stream, it can be used for standard BPSK modulation. The hardware's capability of splitting the digital stream into two orthogonal data streams by taking its odd and even numbered bits can be used as the message generator for QPSK modulation.

The application of this device for BPSK modulation can be described as follows:

The digital output of the hardware $d(t)$, where $d(t) \in \{-1, 1\}$, is used to modulate a carrier signal $S_{tc}(t) = \cos(2\pi f_c t)$. The resulting transmitted BPSK signal is $r(t) = d(t)\cos(2\pi f_c t + \theta_m)$, $\theta_m \in \{0, \pi\}$. This signal is fed into the demodulator, together with a receiving carrier reference signal $S_{rc}(t) = B\cos(2\pi f_c t + \theta_\tau)$, where B is the reference signal amplitude, and θ_τ is a constant phase component. By mixing (multiplying $r(t)$ and $S_{rc}(t)$), the following result is obtained:

$$y(t) = \frac{d(t)B}{2}\cos(\theta_m - \theta_\tau) + \frac{d(t)B}{2}\cos(4\pi f_c t + \theta_m + \theta_\tau) \quad (2)$$

Higher ordered terms can be filtered out using a low pass filter, and the resulting term $\frac{d(t)B}{2}\cos(\theta_m - \theta_\tau)$ is the desired demodulated original message signal, when θ_τ is sufficiently small. This can be achieved by using a PLL circuit to reduce the phase error. The demodulated signal can then be fed into our designed receiver part for DAC. A low cost BPSK demodulator is presented in Benzel[6].

QPSK is a digital modulation scheme where a binary bit stream is divided into an in-phase stream, $d_I(t)$, and a quadrature stream, $d_Q(t)$. Each stream is composed of the odd bits or the even bits of the original message signal, respectively. Each stream assumes half of the data rate of that of the original message signal[7,8,9]. A typical QPSK waveform is achieved by amplitude modulating the in-phase and the quadrature data streams onto the cosine and sine functions of a carrier waveform, and can be mathematically expressed as follows:

$$x(t) = \frac{1}{2}d_I(t)\cos\left(2\pi f_c t + \frac{\pi}{4}\right) + \frac{1}{2}d_Q(t)\sin\left(2\pi f_c t + \frac{\pi}{4}\right) \quad (3)$$

Using our hardware in its QPSK mode, the above data streams $d_I(t)$ and $d_Q(t)$ are obtained. An IQ modulator, such as Minicircuits' ZFMIQ-10M, can be used to generate the above QPSK signal by feeding $d_I(t)$, $d_Q(t)$, and a carrier $S_c(t) = \cos(2\pi f_c t)$ into its I, Q, and LO (local oscillator) ports, respectively. The resulting signal is the transmitted signal and can be demodulated using Minicircuits' ZFMIQ-10D. Detailed procedures of modulating and demodulating BPSK and QPSK signals using Minicircuits' components can be found in [2] and [4]. It is worth noting that the ZFMIQ-10D demodulated outputs contain higher ordered frequency terms. Indeed, when a received signal is mixed with the demodulator LO, simple mathematical manipulation shows the demodulator outputs can be expressed as:

$$I - Channel : x_I(t) = \frac{1}{2}d_I(t) + \frac{1}{2}d_I(t)\cos(4\pi f_c t) + \frac{1}{2}d_Q(t)\sin(4\pi f_c t) \tag{4}$$

$$Q - Channel : x_Q(t) = \frac{1}{2}d_Q(t) - \frac{1}{2}d_Q(t)\cos(4\pi f_c t) + \frac{1}{2}d_I(t)\sin(4\pi f_c t) \tag{5}$$

The higher ordered terms need to be filtered out by a lowpass filter before the baseband signal can be obtained. The obtained $d_I(t)$ and $d_Q(t)$ are then fed into the inputs of our designed receiver circuit. This circuit will synchronize the two data streams, and reassemble them back into the original message stream, and convert the digital signal back to its original analog form. There are three main advantages of using this hardware for BPSK and QPSK modulation experiments:

(i) The designed hardware simplifies the experimental process yet preserves advantages of the systems approach as described in [1]. In the systems approach, students are required to build each individual circuit and check its functionality. It is proven to be difficult for ECET students to accomplish this in a 2-hour block of time. This hardware allows students to connect experiment modules relatively quickly and thus gives students more time to check the functionality of each module.

(ii) The device allows students to better understand modulation and demodulation processes by taking a step-by-step approach. In the case of QPSK, the students can monitor the process and the correctness of generated digital streams before modulation process occur, and can observe the process of filtered signals, the synchronized signals, and the assembled original data stream.

(iii) The device helps students understand the importance of baseband signal synchronization. Traditionally, we stress on the synchronization of RF carriers. In the case of QPSK, the two demodulated streams may not be synchronized after wave-shaping. Our designed hardware incorporates baseband synchronization circuits before reassembling the streams. Using this circuit, students can monitor the synchronization process and learn the adverse effects of reassembling unsynchronized streams (higher Bit Error Rate).

Conclusion

We have discussed results of a senior design project to implement the baseband portion of a communication system. In addition, we have proposed topical discussons and projects that can be introduced into Digital Circuits, Digital Signal Processing, and Electronic Communications courses to enhance student comprehension of the design aspects of communication systems. In a Digital Circuits course, students can implement the hardware design in a hardware description language such as VHDL. This will allow students to simulate, test, and debug their software design on a PC. Students can load and execute their design on a CPLD development board. This exercise provides students with a greater understanding of the design process. In a DSP course, students can test the hardware with DC and sinusoidal input signals to observe and understand the processes of A/D and D/A conversion. Students can also develop software to provide QPSK modulation and demodulation on a DSP chip using as input the data streams provided by the baseband transmitter. In an Electronic Communications course, this hardware can be used with an IQ modulator and demoduluator, such as the Minicircuits' ZFMIQ-10M and ZFMIQ-10D, respectively, to allow students to construct, test and analyze a QPSK data communications system.

References

1. Zhang J. Z., Burbank K. and Adams R., A Systems Approach to Teaching ”Introduction to Electronic Communications” for ECET Students, *Proceedings of the 2004 American Society for Engineering Education Annual Conference & Exposition, ASEE, June, 2004*.

2. Zhang J. Z., Burbank K., Adams R., A Modular Approach to Teaching ”Wireless Communications and Systems” for ECET Students, *Proceedings of the 2005 American Society for Engineering Education Annual Conference & Exposition, ASEE, June, 2005*.

3. Zhang, J. Z., Adams R., Burbank, K., Theory, Practice, and Systems - A New Approach to Teaching Electronic Communications with Matlab®. *International Journal of Engineering Education. Vol. 21, No. 3, 2005* .

4. Zhang, J. Z., Adams R., Burbank, K., The Design of RF Labs using Mini-Circuits Modules to Improve the Quality of Teaching in a Course on Wireless Communications and Systems. *World Transactions on Engineering and Technology Education. Vol. 4, No. 1, pp. 39-42, 2005* .

5. Burbank, K., QPSK Modulator: A Design Example for EET, *Proceedings of the 2003 American Society for Engineering Education Annual Conference & Exposition, ASEE, 2003*.

6. Benzel, D. M., A Small, Low Cost BPSK Demodulator, *RF Design, April 1998.*

7. B. Sklar, *Digital Communications: Fundamentals and Applications*, Second Edition, Prentice Hall PTR, Upper Saddle River, New Jersey, 2000.

8. Proakis, J. G., *Digital Communications*, Third Edition, McGraw-Hill, 1995.

9. Benedetto S. and Biglieri E., *Principles of Digital Transmission with Wireless Applications*, Kluwer Academic / Plenum Publishers, 1999.