AC 2012-4244: INTEGRATION OF NATIONAL INSTRUMENTS MULTISIM AND MATHSOFT MATHCAD INTO A DIGITAL COMMUNICATION TECHNOLOGY CURRICULUM

Dr. David Border, Bowling Green State University

David A. Border, Ph.D., has a principle interest in electronic information systems. This field includes digital communication and networking and intelligent networked devices. Current work includes wireless sensor networks. Prior research included work on signal bandwidth compression and signal specific data encoding techniques. Technology application interest includes networked systems. Typical teaching duties include junior and senior-level courses in the Electronics and Computer Technology (ECT) Program. Within this course set are the curriculum's networking and communication courses. As is true with his ECT faculty colleagues, Border supports the program with teaching assignments, as needed, in freshman and sophomore-level courses offerings. Examples of these include the sophomore level electric circuits and digital electronics courses. Border teaches a digital communication graduate course within the Ph.D. Consortium Technology Management Program, as well as other graduate-level courses at BGSU.

Integration of National Instruments Multisim and Mathsoft Mathcad into a Digital Communication Technology Curriculum

XXX NAME HERE XXX

**XXX PROGRAM NAME HERE XXX
XXX DEPT. NAME HERE XXX
XXX UNIVERSITY NAME HERE XXX
XXX CITY STATE POSTAL HERE XXX**

Abstract

Technology program texts that seek to teach digital communication fundamentals follow basic developments that can be modeled in laboratory classes using computer-based electronics laboratory simulators and computer-based symbolic mathematics systems. For technology programs, this is particularly important as the laboratory work reinforces foundational data communication time domain and frequency domain concepts that may not be readily observed from lecture-only courses. Further, these labs allow students to validate concepts seen in lecture, homework and reading assignments. Additionally, these labs allow students to use digital devices found in earlier course work, such as digital logic. Finally, these labs are meant to complement lectures and not meant to replace laboratory experiments involving actual laboratory digital communication hardware. It is acknowledged that the presence of hardware in the laboratory is vital to the mission of technology education.

National Instruments Multisim is the chosen electronics laboratory simulator for this work. Of special interest in the design of the labs are Multisim's high order capabilities. These allow students to create system level solutions within the setting of a two hour or three hour time period. Mathsoft's Mathcad is the chosen computer-based symbolic mathematics system. Of value to the lab setting, Mathcad is user-friendly and allows the user to setup and complete work within a short time. It preserves the mathematical context of the user's work through its rich symbolic-graphics capabilities. These advantages of Multisim and Mathcad use are discussed in the paper. Finally, it is shown that both Multisim and Mathcad, through print functions and screen captures allow the student to produce notes and illustrations that will complement their notes from lectures and their textbook contents (aside from actual lab report generation).

This paper will detail the laboratory solutions that have been developed to address such topics as signal representation in the time and frequency domain, effects of noise and attenuation on digital signals, channel capacity and bandwidth, spectral and power efficiencies, bit error rate and power efficiency, digital encodings and their time domain and frequency domain characteristics, analog encodings of digital data, analog-to-digital conversion and quantization effects and remedies, and spread spectrum encoding-decoding techniques. The solutions will be framed within the context of illustrating fundamental digital communication principles by use of Multisim and Mathcad. Features and tools of Multisim and Mathcad that are particularly helpful to students will be discussed on an individual and collective lab basis.

I. Introduction

The digital communication technology curriculum can require expensive laboratory hardware. In this context, complementing the lectures with simulation based laboratories can be very useful. In and of themselves, computer simulations are well known and well used in a number of industries. The presence of the computer in the laboratory as an integral part of experimentation and simulation is established¹ in an engineering curriculum. It has been found that such simulation based electronics labs prepare students well for examinations². When they are compared to similar assignments, which are hardware based, these so called, “Inquiry Empowering Technologies” are acknowledged for their ability to reduce time spent in lab³.

The maturity, functionality, ease of use, and level of adoption of two simulation products, Multisim^{4,5,6,7} and Mathcad^{8,9,10}, make them of particular interest for adoption into laboratory work. In this work, a number of simulations have been created to help students better understand principles within digital encoding topics. The simulations are meant to be rapidly assembled, debugged and analyzed. Text and figures that discuss and illustrate the use of these simulations will be shown in a subsequent section.

Student Preparedness

Anecdotally it has been reported that current students are much more comfortable with the use of computer-based symbolic mathematic systems and electronics laboratory simulators than students of a number of years ago. If the anecdotal observation is valid, it could be attributed to improvement in the software applications themselves, and/or it can be attributed to the increasing computer skills of students. Unlike students of previous generations that gained computer proficiency through text-based instruction approaches, including command line interfaces, student computer based work have been and are dominated by graphical user interfaces (GUI). Students are primarily invested with GUI skills developed through the use of office productivity suites. In addition, to office productivity suite work, many electronics technology program students have developed skills from their exposure to computer-aided drafting design software applications, computer programming development environments (e.g. Visual Studio) and programmable logic controller software (e.g. RSLogix5000). The graphical user interface design of both Multisim and Mathcad have commonalities with these and other GUI applications. They provide a top tier of pull down (or pop up) menus and generous workspace (worksheet) below. They employ a standard “point and click” navigation that allows for file management, editing management and help inquiries. Additionally, when working in the workspace (worksheet) standard practices such as, drawing a rectangle about a graphic object and moving it or deleting it, are supported by both applications. Common workspace keyboard short-cuts used in the two applications are already known by many students and allow for rapid performance of a number of basic tasks, such as opening a file, saving a file, printing, making selections, cutting, copying, and pasting.

Graphics in Multisim

In the context of the Multisim-based laboratory work developed for use in teaching digital communication principles to technology students, the chief graphics feature items used are shown in Table 1. All items listed originate in toolboxes, or toolbars, or in a database that can be searched through a component placement dialog. Once the items are “dragged” onto the

workspace, a dragging feature allows wires to be pulled out from one object and attached to another object. Initially, the wire path is automatically determined by Multisim. However, wire paths may be adjusted to suit the needs of the user.

Inputs ↓	Outputs ↓	Model Components ↓
A.C. signal source	Oscilloscope	Individual Logic Gates
Function generator	Spectrum Analyzer	High level functions, such a summer, multiplier, hysteresis block, transfer function block
Bit pattern generator (the “Word Generator”)	Logic Analyzer	Electrical and Electronic components, such as Resistors, Capacitors, Inductors, and Operational Amplifiers.
	Bode Plotter	

Table 1. Chief Multisim Inputs, Outputs and Components Features used in this Work

Graphics in Mathcad

In the context of the Mathcad-based laboratory work developed for use in the digital communications lab, the chief graphics feature items used are shown in Table 2. All items listed originate in toolbars, or through the “insert” pull down menu. Once the items are inserted onto the worksheet, or created with keyboard shortcuts, and arranged into equations and expressions, the worksheet is evaluated top to bottom and results are generated.

Inputs ↓	Outputs ↓	Components ↓
Built in sinusoidal functions	X-Y Plot	Equations, expressions and functions
	Evaluation operation (i.e. =)	Programming operators
		Built-in Functions such as the FFT

Table 2. Chief Mathcad Inputs, Outputs and Components Features used in this Work

Frequently used Simulation Features

In digital encoding work, such as Manchester encoding, the Word Generator can be used to generate both the binary data to be encoded and the clock signal to trigger a flip-flop. In Offset-

QPSK (OQPSK) simulation, the Word Generator is also useful. The Word Generator can be quickly configured to provide both the non-offset bit stream and the offset-bit stream. Most Multisim simulations mentioned in the next section make use of the Word Generator.

The Multisim Spectrum Analyzer appears frequently in the digital communication simulations. Its frequency range is selectable, amplitude range selectable, and most importantly, its frequency resolution is selectable. Due to computational limits programmed into the instrument, choosing a large frequency range inhibits the ability of the analyzer to resolve the analysis into small frequency increments. However, this limitation can be circumvented by using smaller frequency ranges with greater resolution in a piecemeal fashion so to capture a larger frequency range.

Multiplication and addition (summer) functions are not typical in electronic circuit simulations but are present in the Multisim toolbox. They appear in many of the Multisim digital communication simulations. Multiplication functions appear in simulations since they provide a quick method of performing signal modulation and/or signal demodulation. Addition (summer) functions appear in nearly all simulations since they provide a quick method to add signals. Also, summers, through their “properties box,” can also be configured to provide signal voltage gain and/or signal voltage offset. The summer object is pre-configured as a three input device.

The use of a Hierarchical Block (HB) appears in many Multisim digital simulations. It allows the student to a handy way to organize the work of a lab into sub-circuits and facilitates systematic troubleshooting. It allows for quick replication of work common to the whole simulation. Finally, it allows students to reuse portions of work that are common between multiple laboratory assignments.

II. Curriculum Laboratory Experiments

Within a technology digital communication curriculum, five areas of laboratory study were chosen to be investigated on the computer using either Multisim or Mathcad. They were (1) Transmission Lines, Channel Capacity and Bandwidth, (2) Noise Suppression, (3) Shannon Power Efficiency Limit, (4) Digital Encodings, (5) Analog-to-digital Conversion and (6) Parity Generation and Checking.

- Transmission Line

A conceptual point for starting a discussion of data transmission is the source-channel-receiver block diagram. When simplified to a binary source and noiseless channel, students can be moved steadily through an introductory lab on data transmission early in a semester. Building the model allows students to learn and exercise basic skill sets found in Multisim and Mathcad work.

Mathcad use allows quick calculation of the model's length dependent discrete component values and quick computation of channel capacity (2B) once the absolute bandwidth has been estimated in Multisim. It also allows for quick computation of shunt capacitive reactance (etc.) at different operating frequencies. Multisim allows students to estimate channel bandwidth and subsequently to view and compare received waveforms over various distances.

Shown in Figure 1 is the transmission model set up for bandwidth estimation. The “Bode

Plotter” instrument of Multisim is used to plot the frequency response. Students are told to assume the bandwidth is to be measured from the -3 dB point. Note that a standard A.C. voltage source acts as input to the model. Subsequent tasks in this lab ask the students to study the effect due to various bit rate settings and transmission distances on the shape of the received signal when transmitting a string of 1s and 0s. Either a function generator instrument, Word Generator or clock component can be used to simulate the 1s and 0s stream.

Figure 1. Transmission Line Model Work in Multisim

- Noise Suppression

Low pass filter and Schmitt-Trigger noise suppression techniques are examined using Multisim's built-in Hysteresis component. The component allows quick construction of noise suppression circuits. The Hysteresis block has settable properties that allow the student to examine the effect of hysteresis width and input and output limits on signal restoration. The Tektronix oscilloscope model allows students to use realistic oscilloscopes and examine the noisy input signal and the clean reconstructed data signal. The entire simulation is shown in Figure 2. While Multisim can simulate various noise sources, this simulation uses an A.C. voltage source to insert a simple high frequency noise signal.

Figure 2. Noise Suppression Work in Multisim

- Shannon Power Efficiency Limit

Mathcad's ability to mimic the natural appearance of math formulas and relationships allow the construction of a laboratory assignment that walks students through the derivation of the Shannon Power Efficiency limit. The students are prompted through a Mathcad lab exercise, stepwise in design, to derive the limit. The use of “non-evaluated” expressions in the symbolic mathematics system is a key to the process. Once the limit is derived, the “spectral efficiency versus power efficiency” curve can be plotted quickly. This curve, when paired with BER vs E_b/N_0 graphs, becomes an important tool to select better digital encoding techniques that seek to meet a particular design constraint, such as low power constraints or the needs of higher data rate systems. A portion of the work is shown in Figure 3. The figure shows the derived limit (a non-evaluated expression), the writing of the limit as a function, and the Mathcad plot of the limit in decibels.

Figure 3. Shannon Limit Work in Mathcad

- Digital Encoding

- a) Non-phase, non-frequency Based Encoders

Use of Multisim to built digital encoding schemes such as NRZL, NRZI, MLT-3, AMI and Manchester provide provides an opportunity for students to build and analyze encoders, rapidly. Such solutions make use of high level devices such as the Word Generator and a few additional devices such as logic gates and Flip-Flops. Using consistent input bit rates and input voltage levels throughout all solutions, allows students to make spectral comparisons between encodings.

- b) Phase Encoders

BPSK, QPSK and OQPSK can be constructed using a set of Multisim components. In order to observe the behavior of BPSK, QPSK and OQPSK, time domain and frequency domain analysis is done for a number of cases, such as a pseudo random sequence of input bits and a worse case

pattern of 1s and 0s. When the Word Generator is in use, as in the QPSK lab shown in Figure 4 below, the students must be careful to condition their word generator signal to the desired levels. The word generator produces pseudo real life signals that have the low logic level at 0.5 volts and the high logic level at 4.5 volts. In the figure, Multisim's summer are used to interface sections of the simulation which expect different voltage levels. In the QPSK simulation the 4.5/0.5 volt bit levels are adjusted to $+1/-1$ volt. The $+1/-1$ volt levels modulate the carriers. Also, note the “carriers” block in the figure. Named a Hierarchical Block in Multisim work, such sub-circuits are treated much like a subroutine. They receive inputs from the main calling circuit and return outputs. The number of inputs and outputs are configurable. In the work below, the carrier block is configured as a simple output-only block. The carriers are within the block. Within the block, two carrier components are used in quadrature. If a more “realistic” simulation is desired, a single carrier with a 90 degree phase shifter can be used in the block. Note that Figure 4 includes objects not seen in Figure 1 and Figure 2, these are the multiplier component used for modulation. The digital logic instrument is used to examine the split input bit stream. The spectrum analyzer instrument is used to look at the output QPSK spectrum.

Figure 4. QPSK Work in Multisim

c) Frequency Shift Encoders

ASK and BFSK can be constructed using a modest set of Multisim components. As with non-frequency based encoders, the input bit rate and input voltage levels are kept at the same levels for the work so that results can be compared with other simulations. Simple logic gates are used to decode the Word Generator's bit stream and cause conditionally either a 1 volt signal or a 0 volt signal value as input to a multiplier. The other input leg of the multiplier input is attached to a carrier signal. The output of the multiplier will either be zero or a carrier value. In the case of ASK, only one multiplier component and one carrier component is needed. In the case of BFSK, two multipliers component and two carriers components are needed, along with a summer component to sum the multiplier outputs. As with the QPSK simulation oscilloscopes and spectrum analyzers are used to study the operation of the encoder.

- Frequency Hop Spread Spectrum

In this lab work, separate HB blocks are made for the major sections of the FHSS system. Blocks are created for the FSK modulator, the Channel Table and the Frequency Synthesizer. The Word Generator provides both the binary data and the pseudo-random numbers (or PN bits). Students are encouraged to find and use a pseudo-random binary Java Applet number generator to create the PN bit pattern. Alternately a coin flip can be used to create the PN pattern. The narrow-band FSK signal is created using a FHSS MFSK logic block followed with the MSFK generator block. In Figure 5, a simulation of a four-level MSFK slow hop FHSS system is shown. A four-channel hop scheme is provided by the 2 bits assigned as PN bits. The completed FHSS system allows students to observe transmitted data frequency hopping.

Figure 5. FHSS Work in Multisim

Both slow hop and fast hop are modelled. In the figure shown, the QPSK output spectrum analyzer by chance shows both the old (decaying) and new data value (enlarging) just as it hops between frequency levels. The second spectrum analyzer shows the data value in the narrow-band stage prior to the high frequency modulation section.

- Analog-to-digital

a) Sampling and Reconstruction

Mathcad is used to show the theory of PAM operation when sampling a periodic function. PAM sampling results are varied by changing the sampling interval and sample width constants. The sample interval is determined by the number of conditional “if” statements in the overall programming statement. In Figure 6, five conditional if statements are used, so the sample rate is 5 samples/cycle. The programming statement builds the value of a time domain function named “sampled_wave(t).” Each statement is triggered when the periodic function is at a particular phase angle (measured in degrees). If additional “conditional if statements” are added to the programming statement, the numbers of samples per cycle are increased. If statements are subtracted, then the number of samples per cycle are decreased.

Figure 6. Programming in Mathad to Create Sample Signal

During the program statement evaluation, if all “if statements” conditions prove false, a zero is inserted into sampled_wave(t). The sample width shown in the figure is intentionally large to

illustrate the resulting peak amplitude decay along the frequency axis.

Mathcad's intrinsic FFT function provides the spectral analysis. It is shown in Figure 7. The first task of this section is to setup the FFT's input vector. The setup of the input vector can be used as an opportunity to teach students the basics of FFT theory, or it can be presented as "given" within the context of the lab.

Figure 7. FFT Work in Mathcad

Following the Mathcad sampling exercise, Multisim is used to both sample and reconstruct a trivial sinusoidal function. The Multisim toolbox contains a simple voltage controlled switch component that allows for quick construction of the sampling circuit. A low frequency message signal source is used along with a very high sampling rate circuit (far above the Nyquist rate). This allows students to make use of a simple reconstruction filter (built from Multisim's transfer function block feature or a simple RC LPF circuit) to obtain an acceptable estimate of the original signal. Use of Multisim's oscilloscope instruments and spectrum analyzer instruments allow students to analyze the whole process of signal sampling and signal reconstruction.

b) Undesirable Quantization Truncation Effects and Dithering

Mathcad is used to show that truncating values or rounding of values, such as could be found in the sampling and quantizing of audio signals or picture signals can result in the creation of artifacts in the quantized signal that are periodic. Such error-based periodic "noisiness" can be noticeable to the ear (in audio processing) or to the eye (in picture processing). Obtaining a new stream by summing random values into the output sample stream to create a "dithered" sample stream removes the periodic artifacts. In Figure 9, this process is shown mathematically using

Mathcad. In one case, students sample, quantize (truncate) and save the values. In the other case, students sample, quantize (truncate) and then dither using the “rnd” random function. The quantizing error for each case is then plotted. By visual inspection, the periodic nature of the non-dithered error signal is seen. Likewise, by visual inspection, the lack of periodicity in the dithered error signal is also seen. In the laboratory work, the students are told to perform FFTs to verify these time-domain observations.

Figure 8. Dithering Work in Multisim

- Parity Generation and Checking in Triangular Form

Students are told to build a 10-bit data generator using the Word Generator and send the data in triangular form with parity. The data is arranged in triangular form with 4-bit parity. As with the FHSS laboratory work, Hierarchical Blocks are used to organize and simplified the process of simulation. In Figure 9, the Multisim triangular encoding solution is shown. A sorter block at the transmitter and a sorter block at the receiver (identical blocks) sort the 10-bit data into the correct bit patterns expected when using this size triangular parity encoding form. The parity circuits are assembled into a Hierarchical Block. The HB subcircuit is replicated to provide for triangular form parity generation. A total of 5 HB parity generator blocks are used at the transmitter and a total of 5 HB parity generator blocks are used at the receiver. All blocks are identical.

The 5 parity bits created at the transmitter and the 10 data bits are input to the receiver section. The receiver's 5 XOR-gates compare transmitter parity and receiver parity results and output them to five LED-like (Multisim “probe” component) components. This section displays the

syndrome value. A “NOT” gate component is introduced in data bit 10 to show the effect of a one-bit error on the syndrome. Students are told to relocate the NOT gate elsewhere in other data lines or parity lines to illustrate other syndromes.

Figure 9. Parity Encoded in Triangular Form Work in Multism

III. Assessment

Intentional lab assessment work is done formally through examination of submitted student laboratory work, grading of laboratory work content, grading of written examinations and reading the student laboratory feedback solicited at the end of the semester. Additionally, informally collected feedback is provided by students throughout the semester. There are a number of positive outcomes indicated through these assessments.

- Simulations are built in the allotted time. Most students obtain the expected simulation results. Many students report that troubleshooting the GUI programs is less tedious than troubleshooting hardware.
- Students exhibit a better grasp of frequency domain concepts. Examinations confirm this conclusion.
- Students have their knowledge from prior courses, such as digital logic and analog communication systems, intentionally refreshed and reinforced.
- Students are able to prepare higher quality laboratory illustrations by incorporating their GUI work into office productivity suite word processors (comparable with the figures

shown in this work).

There are certain negative work aspects noted during the review of the assessments. These include:

- Student carelessness observed during the labs. For example, it is noted that some students are less adept at making GUI-based Multisim wired connections (which can result in floating elements). An example of common errors seen in Mathcad: some students fail to use the correct “equality” symbol. Equality can be found in the act of assigning a value to a function or variable, or it can be found in a “non-evaluation” form of equality, or it can be found in the act of “reporting” the value of a function or variable. Using the correct equality symbol is critical.
- Unevenness in laboratory computer responsiveness. In practice having ten computers behave exactly the same (despite the fact that they have identical hardware and operating systems) is found to be quite difficult. On occasion some computers lockup and some applications freeze. Other occasional problems besides these have been noted.
- In Multisim, scale and offset operations and range computations confuse some of the students. Forgetting to scale or offset will cause unexpected results. Much like the real life issues encountered when interfacing two different voltage level standards, say TTL to RS-232, interconnecting Multisim devices such as the Word Generator (TTL) to special stages must be done correctly. In some labs, students must manually scale and offset Word Generator output, or alternately, use a buffer component to do the task. Range computations are most often needed when using the instruments such as the Spectrum Analyzer. Improperly setting instrument ranges will cause unexpected results.

IV. Conclusions

For three years, these Multisim and Mathcad based laboratories have been used to illustrate concepts of digital encoding and digital communication in a senior level course. The use of fewer components, through the use of higher level components, in practice, has allowed most all of the students to accomplish a given simulation within one lab session. With careful troubleshooting, important lecture concepts are successfully validated by students. The mechanics of using these GUI applications themselves is considered “user friendly” by the students and help them to accomplish labs within a reasonable amount of time. However, experience has shown that since the simulations are executed on Windows XP computers within a laboratory shared with a wide variety of actively used software applications, unexpected interruptions due to computer freezes (etc.) can occur and spoil an unsuspecting student's work.

Future work will concentrate on moving the lab work in two directions. One direction is to take a few portions within this material and move them into the program's sophomore year electronics digital logic course. An example would be to use encoding schemes such as AMI or NRZI as the basis for digital logic projects. This will help students better understand the interrelation between areas of electronic technologies. The other direction is to take a few portions within the material used and convert them from basic component solutions to more detailed solutions. One such substitution would be to use a more complex LPF as a replacement to the RC filter in the sampling reconstruction lab. This would reinforce junior-level filter design instruction.

Bibliography

1. Feisel, L.D. and Rosa, A.J., The Role of the Laboratory in Undergraduate Engineering Education. *Journal of Engineering Education*, Vol. 94, No 1, January 2005, pp. 121-130.
2. Campbell, J.O., Bourne, R.J., Mosterman, P.J., and Brodersen, J.A., The Effectiveness of Learning Simulators in Electronic Laboratories. *Journal of Engineering Education*, Vol.91, No.1, 2002, pp. 81–87.
3. Hofstein, A., and Lunetta, V., The Laboratory in Science Education: Foundations for the Twenty-First Century. *Science Education*, Vol. 88, Issue 1, pp. 28-54.
4. <http://www.ni.com/academic/multisim.htm>.
5. Yildiz F., Pecen R., Zora A., Utilizing Advanced Software Tools in Engineering and Industrial Technology Curriculum. *International Journal of Engineering Research and Innovation (IJERI)*, Online ISSN: 2152-4165, Volume 2, Number 1, Spring/Summer 2010, pp. 12-22.
6. Wang G., Principles and Practices: Multisim in Teaching Digital Systems Design. *Proceedings of 2006 ASEE Illinois/Indiana and North Central Conference*, March 31- April 1, 2006, Fort Wayne, In.
7. Noga, K.M. And Radwanski, M., Using the Virtual Model in Teaching Digital Signal Processing. In *Proceedings of EIAT/IETA*. 2008, 195-200.
8. <http://www.ptc.com/products/mathcad/>
9. Domnisoru, C., Using Mathcad in Teaching Power Engineering. *IEEE Transactions on Education*, Vol 48, Issue 1, February 2005, pp. 157-161.
10. Baltzis, B.B., Proposing a Mathematical Software Tool in Physics Secondary Education. *Acta Didactica Napocensia*, Vol 2, No 1, 2009, pp. 9-18.