USING DSP HARDWARE TO TEACH MODEM DESIGN AND ANALYSIS TECHNIQUES

Thad B. Welch, Robert W. Ives

Department of Electrical Engineering
U.S. Naval Academy, MD
t.b.welch@ieee.org, ives@usna.edu

Michael G. Morrow

Department of Electrical Engineering
University of Wisconsin, WI
morrow@ieee.org

Cameron H. G. Wright

Department of Electrical Engineering
University of Wyoming, WY
c.h.g.wright@ieee.org

ABSTRACT

While many communication topics are difficult for undergraduate students to internalize, demonstrations and laboratory experiences have been shown to greatly stimulate the learning process. This paper describes a highly successful combination of theory, demonstrations, lab exercises, and real-time DSP experiences using MATLAB and the Texas Instruments C6711 digital signal processing starter kit. This approach, when also combined with state-of-the-art measurement equipment, has proven highly successful in reinforcing modem design and analysis techniques.

1. INTRODUCTION

This paper describes the addition of digital communication transmitter capability to the winDSK6 program [1, 2]. This Microsoft Windows® application controls a Texas Instruments (TI) TMS320C6711 or TMS320C6211 digital signal processing starter kit (DSK). The C6x DSKs incorporate a much more potent processor and greater memory compared to earlier DSKs. This new DSK also ships with TI's Code Composer Studio (CCS), a complete, integrated production-grade code generation, debugging, and analysis environment. While this greatly enhances the ability of skilled programmers to implement significant DSP algorithms in a laboratory setting, it also makes it more difficult to perform demonstrations for several reasons. First, the DSK hardware is now dependent on driver files installed as part of CCS, meaning the DSK hardware cannot be used except on machines with CCS installed. This can pose a serious burden for educators who want to perform demonstrations in classrooms or laboratories where CCS is not installed. This could be due to a lack of sufficient licenses or to not having time to install and uninstall CCS for a demonstration. A related problem we found is that driver modifications made by TI for revisions to CCS caused compatibility problems. Another issue is related to the extensive professional-grade capabilities of CCS itself: it is a very complex software package, and to the novice it is often overwhelming.

Software support for this project was provided by Agilent Technologies, Inc.

For example, if your pedagogical goal is to get students to experiment with different FIR filter designs and observe the results in real-time, then using CCS for this is overkill and a potential inhibitor to learning. In order to take students on a journey from theory to real-time practice, there needs to be an infrastructure in place to support them and target as many modes of learning as reasonably possible [3]. One very important segment of this needed infrastructure is tools to support faculty demonstrations, student experimentation, and student self-learning. Professional-grade tools such as CCS are too complex to lend themselves easily to this pursuit.

2. A SOLUTION: WINDSK6

To overcome these difficulties, the authors developed a new software package tailored to the TMS320C6711 DSK. After several years of experience with our original winDSK for the TMS320C31 DSK, the new winDSK6 is much improved and more capable, and takes full advantage of the higher performance DSK. The winDSK6 program is a Windows 9X/NT/2000 application that provides an intuitive and easy-to-use interface, and ensures that a student's first experience with the DSK is a positive and motivating one. It makes the DSK hardware much more accessible to students, and facilitates easy-to-use, ready-made classroom and laboratory demonstrations. For simplicity, all application software and DSK code is embedded in the executable file. A help file provides a section on each demonstration that discusses the theory and operation of the application, and context-specific help is available on each application control. To eliminate the requirement to have CCS installed on the machine, a completely new DSK driver was developed that operates under Windows 9X/NT/2000 [1]. The winDSK6 program, written in C++, encapsulates the DSK's physical and logical interface to the host computer. The applications that form the basis of winDSK6 have evolved over time and experience, with the authors' needs in the classroom and laboratory being the motivating force behind the new capabilities in the current version. Individual applications are all

[1]Operation under Windows XP has not been thoroughly tested as of the submission date of this paper.

dialog-based, and perform a similar sequence of operations to execute an application program:

1. Reset and reboot the DSK
2. Download the application software
3. Locate the shared memory block and initialize any data to synchronize the host computer and the DSK
4. Run the DSK application

This all occurs upon selecting an application with a mouse click; to the user it appears to occur instantaneously. Once the application is running, communication via the shared memory block is used to control the DSK application's behavior in response to user input via the dialog window displayed on the host computer. This gives users real-time interactivity and immediate feedback when changes are made on the host computer.

3. DEMONSTRATION APPLICATIONS

The winDSK6 demonstration applications highlight a number of signal processing operations. Nearly all applications require only the basic DSK hardware to operate, although the limitations of the DSK's onboard codec do significantly restrict what can be accomplished. In our experience, we have found that using one of the CD-quality codec daughtercards (available from Texas Instruments, Educational DSP, and other sources) to be much more useful. For this reason, winDSK6 provides support for these more capable analog interfaces for most applications. Applications also include a DSK Settings button that allows for the control of codec functions (such as sampling rate) in real-time while the application is running on the DSK. As shown in Figure 1, the available applications include an oscilloscope/spectrum analyzer, multi-channel signal and/or arbitrary waveform generator, graphic equalizer, audio effects, guitar synthesizer, several filters, DTMF generator, and a multi-mode digital communication transmitter. Selecting the commDSK option from the winDSK6 user interface launches the commDSK application and brings up the window shown in Figure 2.

The remainder of this paper will discuss the features of the new multi-mode digital communication transmitter provided by commDSK along with the hardware and software tools used to analyze the generated signals.

4. COMMDSK FEATURES

As shown in Figure 2, the user interface is divided into three sections that are labeled:

1. Modulation Control

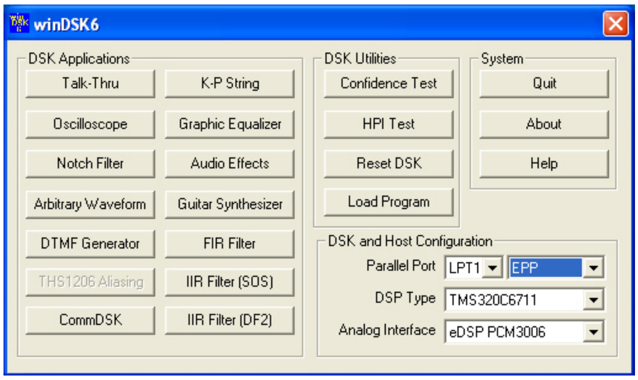

Fig. 1. The main winDSK6 user interface.

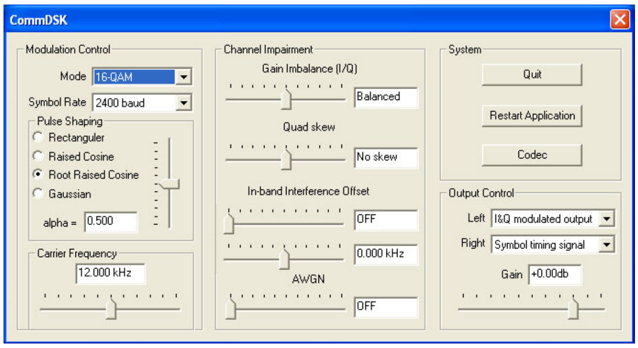

Fig. 2. The commDSK user interface.

2. Channel Impairment
3. Output Control

The Modulation Control section specifies the modulation scheme (Mode), the symbol rate, pulse shaping and the carrier frequency.

1. The six available modulation schemes BPSK, QPSK, 8-PSK, 16-PSK, 8-QAM, and 16-QAM.
2. The available pulse shaping techniques include rectangular, raised cosine, square-root raised cosine, and Gaussian. Where appropriate, the roll-off parameter is variable.
3. Finally, the carrier frequency is variable from 0 Hz to 24 kHz, which is one half of the system's sample frequency.

Although most features of winDSK6 allow for a variable sample frequency, this application currently restricts the sample frequency to 48 kHz. This restriction obviously requires a codec capable of running at this frequency for proper operation.

The Channel Impairment section specifies various effects that can occur in a digital communication system. Any of these effects may be used in combination to simulate as many simultaneous effects as the user desires.

1. The *gain imbalance* slider control varies the ratio of the in-phase (I) and quadrature (Q) channel magnitude ratio (I/Q) from 0.5 to 2.0. This effect turns square constellations into rectangular constellations.

2. The *quad skew* slider control varies the degree of orthogonality between the I and Q oscillators by as much as ± 30 degrees.

3. The *in-band interference offset* slider controls both the offset frequency of an interferer and the magnitude of the interference.

4. The *AWGN* slider control allows additive white Gaussian noise to be added to the signal to allow for variable signal-to-noise ratios or energy-per-bit to noise spectral density ratios to be generated.

The Output Control section specifies which of seven possible output signals will be ported to the left and right output channels and at what level the signals will be generated. The output selection control allows for:

1. I and Q modulated output. This option is used for higher-order modulation schemes where a single signal is desired.

2. I modulated output. This option is used when only the I portion of the modulated signal is desired.

3. Q modulated output. This option is used when only the Q portion of the modulated signal is desired.

4. I baseband signal. This option is used when only the I baseband signal is desired. This signal has its data pulse shaped but it is not mixed with the cosine carrier at the carrier frequency.

5. Q baseband signal. This option is used when only the Q baseband signal is desired. This signal has its data pulse shaped but is not mixed with the sine carrier at the carrier frequency.

6. Symbol timing signal. This signal has a rapidly rising pulse at the center of each symbol. This signal is intended to aid in receiver timing recovery or eye-pattern generation through oscilloscope triggering.

7. No output (off). This option sets the channel output to a value of zero volts.

Finally, the Gain adjustment allows for signal attenuation or gain. This slider control is intended to show the effects of not using the full dynamic range of the digital-to-analog converter (DAC) or saturating (over-ranging) the DAC.

5. SIGNAL ACQUISITION AND ANALYSIS

With this new ability to generate a large number of different digital communication signals, our students are now able to analyze and test these systems using state-of-the-art test equipment and software. Specifically, they gather samples from their DSP-based communication sources using Infiniium oscilloscopes from Agilent Technologies, Inc. These signal samples are then passed to a Windows 2000 based workstation using a general purpose instrumentation bus (GPIB). The workstation is running the vector signal analyzer (VSA) software associated with the Agilent 89601A VSA system and displaying a number of analysis plots in near-real-time. Examples of only a fraction of the displays are provided as Figures 3, 4, 5, 6, and 7. Several of these plots allow for impairment identification that is not possible using traditionally available test equipment.

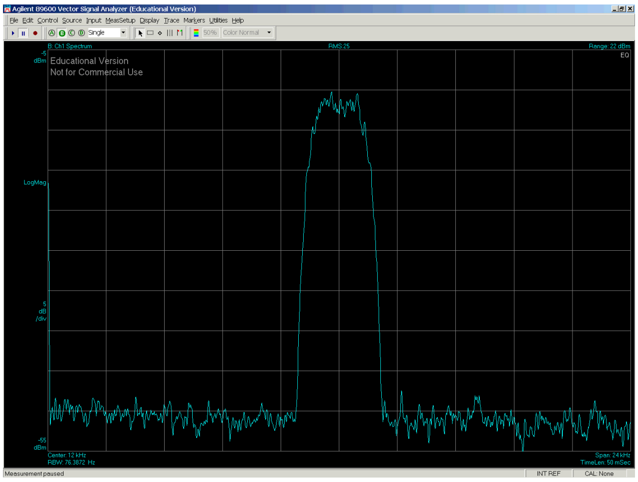

Fig. 3. The spectrum of a 2400 baud, 16-QAM signal, with a carrier frequency of 12 kHz.

With very affordable and easily reconfigurable communication signal sources at every laboratory station and the high performance hardware and software necessary to analyze and test these communication systems for a large number of impairments, our students are now motivated to design, build, and test not only their own transmitters, but their own receivers as well.

6. CONCLUSIONS

We have developed and described a significant enhancement to the winDSK6 program that has allowed for a large number of digital communication signals to be generated both affordably and conveniently. These signals can be intentionally impaired to allow for test and evaluation education with state-of-the-art hardware and software tools. The hardware

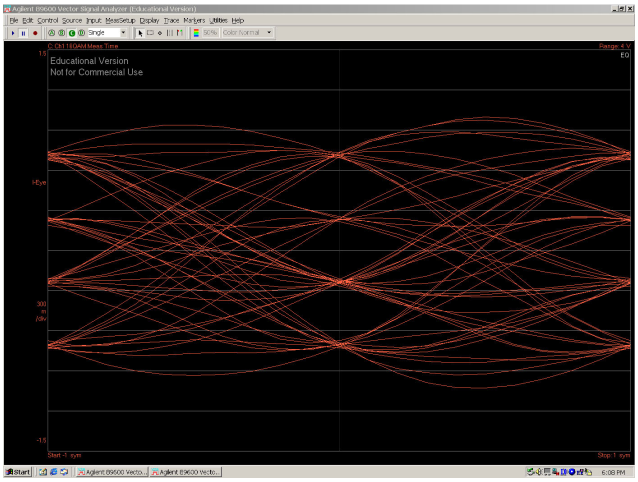

Fig. 4. The eye-diagram of a 2400 baud, 16-QAM signal, with a carrier frequency of 12 kHz.

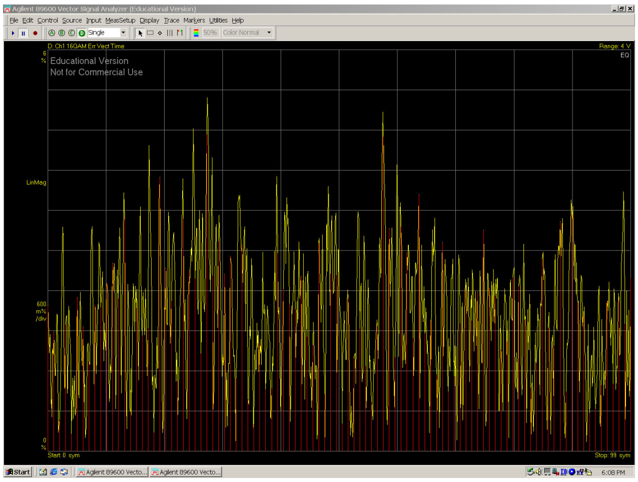

Fig. 6. The error vector magnitude of a 2400 baud, 16-QAM signal, with a carrier frequency of 12 kHz.

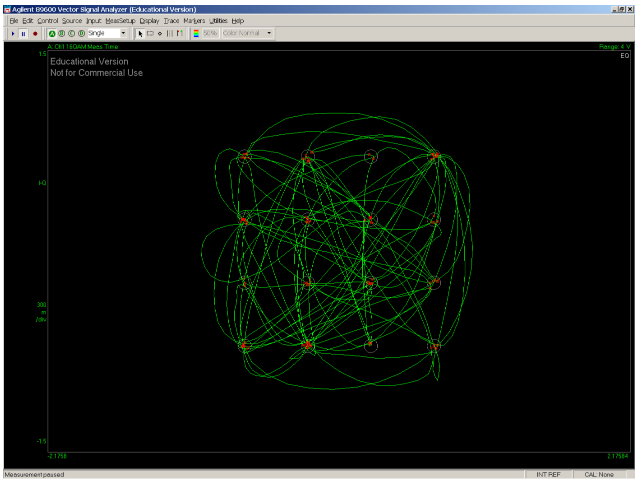

Fig. 5. The constellation or trajectory diagram of a 2400 baud, 16-QAM signal, with a carrier frequency of 12 kHz.

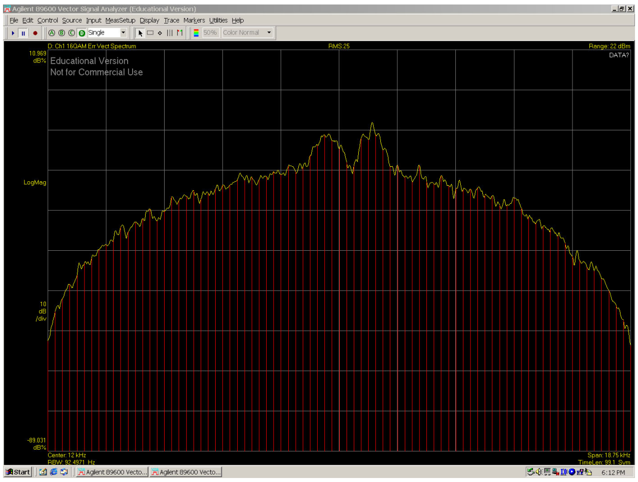

Fig. 7. The error vector magnitude spectrum of a 2400 baud, 16-QAM signal, with a carrier frequency of 12 kHz.

investment required to implement the communication system sources is rather modest, and all of the needed software has already been developed by the authors or is available from Agilent Technologies, Inc.

We freely distribute the winDSK6 software for educational, non-profit use, and invite user suggestions for improvement. See `http://eceserv0.ece.wisc.edu/~morrow/software/`; interested parties are also invited to contact the authors via e-mail.

7. REFERENCES

[1] M. G. Morrow, T. B. Welch, and C. H. G. Wright, “A tool for real-time DSP demonstration and experimentation,” in *Proceedings of the 10th IEEE Digital Signal Processing Workshop*, (Pine Mountain, GA (USA)), Oct. 2002. Paper 4.8.

[2] M. G. Morrow, T. B. Welch, and C. H. G. Wright, “An inexpensive software tool for teaching real-time DSP,” in *Proceedings of the 1st IEEE DSP in Education Workshop*, (Hunt, TX), IEEE Signal Processing Society, Oct. 2000.

[3] C. H. G. Wright, T. B. Welch, D. M. Etter, and M. G. Morrow, “Teaching DSP: Bridging the gap from theory to real-time hardware,” in *Proceedings of the 2002 ASEE Annual Conference*, (Montréal, Québec, Canada), June 2002.