An Introduction to Hardware-Based DSP Using winDSK6

Michael G. Morrow
University of Wisconsin
Thad B. Welch
United States Naval Academy
Cameron H. G. Wright
U.S. Air Force Academy

Abstract

Today there is a global need for engineers who are DSP literate. To help educate the next generation of DSP engineers, several powerful and highly versatile DSP boards designed for educational use are offered. The problem with most of these systems is that they lack user-friendly software to demonstrate the capabilities of the DSP boards. Some of the software packages shipped with these devices have a very steep learning curve, while other packages are still DOS or command line based. Today's students may be hesitant to approach these impediments to learning without significant motivation. To allow a student's first hardware-based DSP experience to be a positive one, software tools must be provided that are affordable, easy to install, attention getting, Windows-based, and feature rich. This paper discusses winDSK6, an object-oriented application program that meets all of these requirements. winDSK6 seamlessly interfaces with and programs the complete TMS3206xxx family of DSP Starter Kits (DSKs). Its features include, talk-through (with full CODEC parameter control), oscilloscope, spectrum analyzer (both the traditional display and waterfalling spectrogram), notched filter, arbitrary waveform generation, DTMF generation (with repeating speed-dial), Karplus-Strong string algorithm, 5 band graphic equalizer, audio effects, guitar synthesizer, and a DSK confidence test. The authors freely distribute this software for educational, non-profit use.

1 Introduction

There is a worldwide need for digital signal processing (DSP) literate engineers that is not being completely satisfied. Despite our best efforts to offer both innovative electrical engineering (EE) and particularly DSP courses that use state-of-the-art software and hardware tools (such as The MathWorks' line of MATLAB products and Texas Instruments' (TI) TMS320Cxxxx DSP boards and associated development tools), EE enrollment continues to decline! If the decline in EE enrollment is a trend that we as professional EE educators wish to reverse (and the authors submit that we do), then a more detailed look at our students is in order.

If we assume that the majority of students who attend universities start their freshman year already knowing what academic major they plan to pursue, then the best we as DSP educators can hope to do is to encourage those students who have already selected EE as their major to study DSP. If this assumption is wrong, then recruiting freshman into EE can remedy the shortage of electrical engineers in general and the shortage of DSP-literate engineers in particular.

However, we believe that both of these courses of action are shortsighted, since the DSP educator waits until the student is already at the university before becoming involved with the student. Earlier intervention must be considered for optimal results. Additional shortcomings of either of these two courses of action include:

- They overlook the need to educate students in other engineering disciplines concerning cross-disciplinary DSP topics, and
- It does not address the real issue of increasing the talent pool from which we as DSP educators can draw our students.

The idea that *DSP isn't just for EEs anymore* is an outgrowth of the realization that topics such as data acquisition (with the subsequent data analysis) and feedback control are vital to most engineering disciplines. Academically, this can be seen from the trend of EE courses for non-EEs that are migrating away from the traditional *DC and AC Circuit Theory* type courses towards courses that include a significant coverage of *Signals and System*s.

While the idea of increasing the talent pool from which we as DSP educators draw our students is by no means new, accepting some degree of personal responsibility for "reaching out" to students that are years away from our university classrooms may be new to some. Some possible forms of outreach may be mentoring, high school career connection days (answering questions such as… "What does a DSP engineer do?"), non-traditional course development, offering continuing education courses, and software development, to name but a few.

Today's student is significantly different from the majority of DSP educators. Putting the "digital divide" aside, the "Nintendo generation" is largely

- computer literate,
- graphically oriented,
- possesses refined hand-to-eye coordination, and
- expects immediate feedback or results.

While at first glance this generation may seem like ideal students, we believe these attributes have led to a great deal of academic impatience. They don't respond well to the slow building-block approach of most engineering programs unless they are highly motivated [1-4]. Since we have a decidedly different type of student today, software tools that are crafted to take advantage of this unusual mix of attributes should be in use around the world.

1.1 Professional Software Tools

Most of these tools are very expensive (very expensive being defined as, you would not expect your students to buy the product for one course). While some companies (e.g. The MathWorks and TI) have long histories of offering deep discounts to educators on their products, the fact

remains that these are *professional tools*, which in general, were not initially designed with education in mind. For the experienced student these tools are like a dream-come-true, in that they have the power that a professional set of tools brings, but at an educationally affordable price! On the other hand, for the inexperienced student, these same tools may be described as too powerful, too steep of a learning curve, too feature rich, confusing, etc…

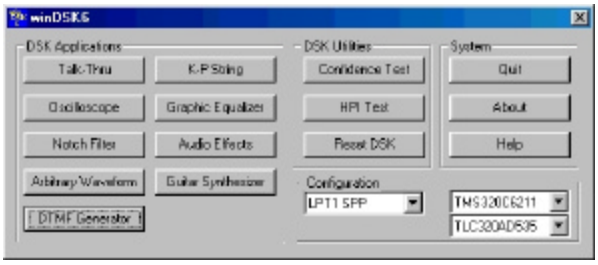
1.2 Educational Software Tools

Most of these tools are inexpensive but usually have one or more of the following drawbacks. They do not have a graphical user interface (GUI), are limited in capability, or despite their educational intent, they are difficult to learn or use.

For both of these broad categories, the Nintendo generation's expectations are not met. What they want is a third category that largely resembles a computer game until they have seen the utility of investing their time and effort to master the subject material or the required software/hardware tools needed to support their study of the subject material. This type of tool may be thought of as a transition tool that facilitates the initial use of one of the other two categories of tools.

It may appear that we are suggesting that we attract students with game-like techniques that obscure the actual effort required to solve the underlying mathematics. This, however, is not the case. Using this type of software is a recognition that today's student has different expectations. They want an "out of the box" experience, one that requires them to take the CD out of the box, install the software, and be experiencing DSP in just a few minutes!

2 One Solution to this Problem



winDSK6 is a Windows 9X/NT application that interfaces directly to the TI TMS320C6X11 DSP Starter Kit (DSK) hardware [5]. It is intended to provide a positive initial experience for students, and to provide additional utilities and functions to make the DSK even more accessible to them. It contains numerous demonstration programs that emphasize audio and visual interaction, including an oscilloscope/spectrum analyzer, graphic equalizer, audio effects, DTMF generator, and arbitrary waveform generator. Help files for each application introduce the underlying theory at work in producing the effect. winDSK6 also provides a DSK Confidence Test, and a Host Port Interface testing and debugging tool.

2.1 Software Architecture and Operation

winDSK6 is a stand-alone application consisting only of the executable and help files. The application is written in C++, and is centered on an object that encapsulates the DSK and its physical interface to the host computer. Communications between the host PC and the DSK are based on the API supplied by Texas Instruments, with the host PC mastering all communications. To provide the most versatility, only a single memory location in the DSK

software, which contains the address of the shared data table that the host accesses, is required to be in a fixed location. Individual applications are dialog-based, and each performs a similar task sequence to execute the appropriate DSK application;

- Reset the DSK, load the DSK application code, and start it running,
- Determine the location of the DSK application's shared data table, and
- Synchronize the host user interface and the DSK state.

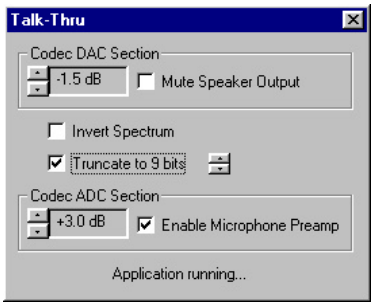
All this occurs immediately upon selecting an application button. Once the application is running, the host computer controls the DSK application's behavior by modifying memory locations within the DSK application shared data table in response to the user changing control settings in the dialog window. This real-time interactivity provides immediate results on the DSK in response to user actions.

3 DSK Applications

The demonstration applications showcase a number of audio signal processing operations. All programs only require the basic DSK in order to operate; no additional circuitry is required. Several applications were in fact added in response to specific requests from professors who needed to demonstrate certain effects. A short theory section in the help files that explains the operation of the application supports each application.

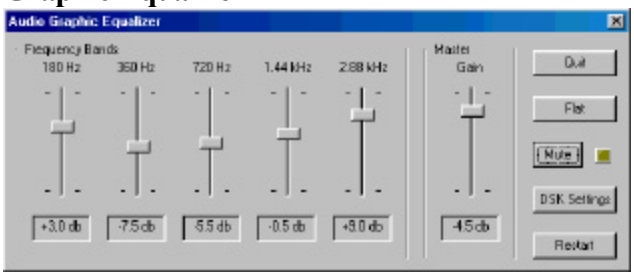
A "DSK Settings" button on most applications allows the user to reconfigure the analog interface circuit on the fly. A "Restart" button allows restarting each application at its default settings. The demonstration applications include;

Talk-Thru



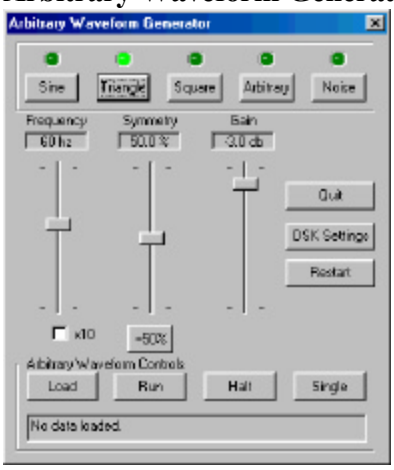
This application takes audio input from the ADC and passes it back out through the DAC. All aspects of the TLC320AD535 CODEC on the DSK can be controlled. The user can also vary the effective number of bits from the full 15-bit range of the AIC down to a single bit, demonstrating the relationship between quantization and signal-to-noise ratio, as well as our complex perception of information from a noisy signal. The concept of spectral inversion can also be demonstrated.

Graphic Equalizer



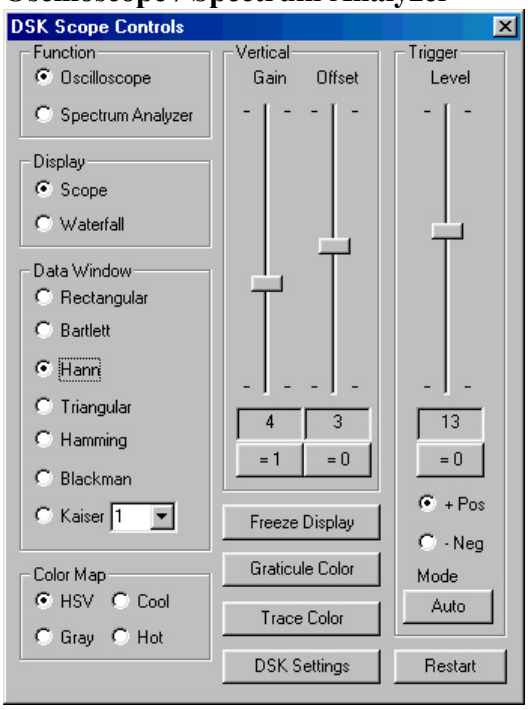
The five-band graphic equalizer uses five parallel 128th order FIR filters, with the gain of each filter independently controlled by a slider control. A master gain control, mute button, and flat button provide additional control.

Arbitrary Waveform Generator

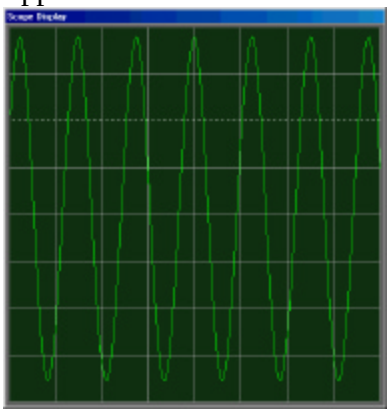


The arbitrary waveform generator generates sine, square, and triangle waves at frequencies between 1 Hz and 1 kHz, with selectable symmetry for the triangle and square waves. A noise generator function is also available. In addition, it can load up to 1 million sample values from an ASCII file and then operate as an arbitrary waveform generator. The data is automatically scaled to the full range of the DAC. Continuous and single pass arbitrary signal generation is supported. A sample chirp file is included with the software package.

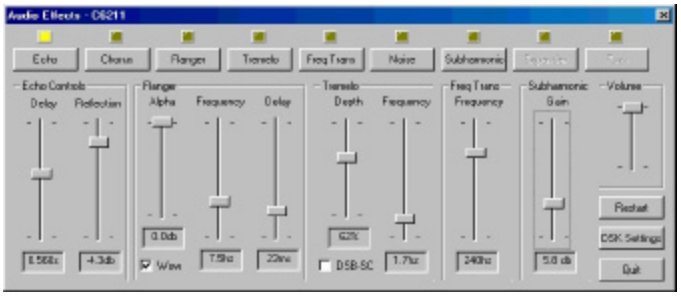
Oscilloscope / Spectrum Analyzer



The oscilloscope / spectrum analyzer can operate as a standard oscilloscope or as a spectrum analyzer. The resizable display window can be viewed as a conventional oscilloscope, or a waterfall display can be selected with various colormaps. For spectral analysis, a number of different data windows can be selected. Standard oscilloscope triggering methods are also supported.

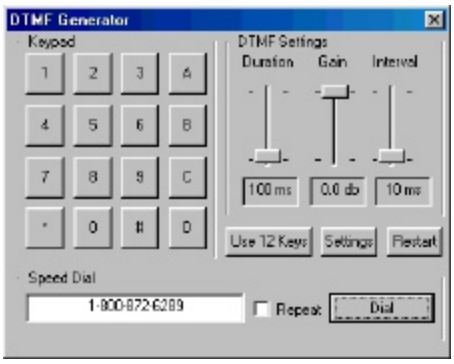


Audio Effects



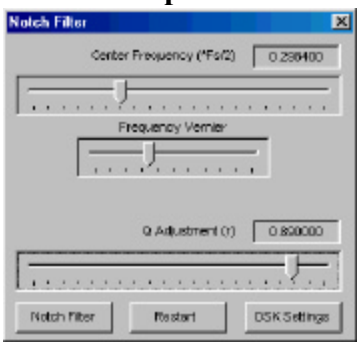
The audio effects application demonstrates a variety of signal manipulations possible using DSP, including echo, chorus, flanging, tremolo (amplitude modulation), noise, and frequency translation. Many of the effect parameters can be varied well beyond the customary values used with music to produce unusual effects.

DTMF Generator



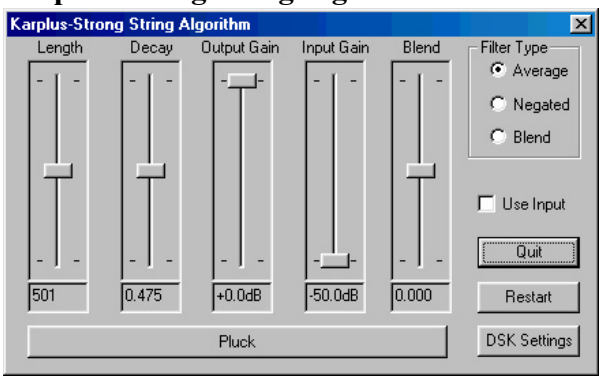
The DTMF generator can generate all 16 DTMF tone pairs. The duration of each tone, as well as the interval between tones, can be adjusted. A 12-key or 16-key keypad can be selected. A speed dial feature operates in the same way as a typical telephone, and includes a continuous repeat function.

Notch/Bandpass Filter



This application implements a single stage IIR filter acting as a notch or bandpass filter. The filter's center frequency and Q can be adjusted.

Karplus-Strong String Algorithm

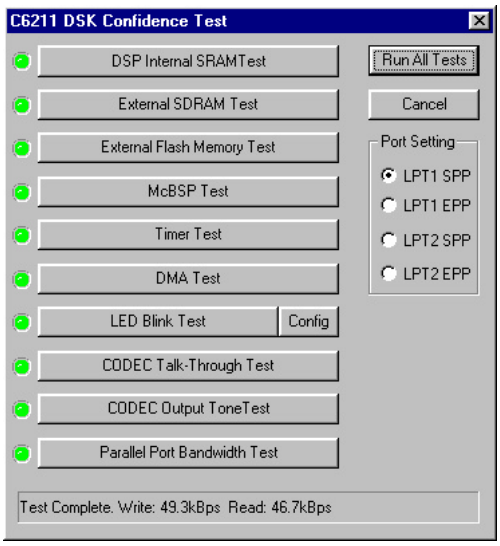


Implements the Karplus-Strong string algorithm [7] to synthesize a plucked string. The delay buffer length, decay coefficient, and the filter type are all user adjustable. Additionally, the ADC input can be processed through the algorithm by using it to excite the string buffer on a continuous basis.

4 DSK Utilities

The DSK utilities provide support for testing and debugging DSK applications. A flash memory programming utility will be added in a future version to facilitate using the DSK in bootable embedded system projects.

DSK Confidence Test



The DSK Confidence Test is an easy way to test the basic functionality of the DSK, providing a graphical interface to the confidence test supplied with the DSK. Visual indicators show test progress and status. Each test may be run individually, or all can run in an automatic sequence. In addition, the DSK Confidence Test also tests and reports the speed of the parallel port connection between the host PC and the DSK in both directions.

Host Port Interface (HPI) Test

The Host Port Interface (HPI) Test application tool includes a program loader, and permits reading/writing selectable memory locations while a DSK program is executing. Memory locations can be selecting by entering the appropriate value, or by picking program labels from a list. The data can be viewed, edited, and written in floating point or integer forms. The HPI Test is particularly useful for modifying memory when reading and writing must be done independently, a function that is not available under Code Composer Studio.

5 Future Enhancements

The most exciting planned enhancement to winDSK6 will be added support for a low cost daughtercard under development that will provide a two-channel analog interface capable of 16-bit operation at 48 KHz. This will greatly enhance the capabilities of the DSK, and open the door to many more interesting applications. Optimized native support for the TMS320C6711 floating point DSK will also be incorporated in future versions. Additional applications and demonstrations will be incorporated as well, and updated versions will be made available as discussed below. Specific suggestions for enhancement should be directed to the authors.

6 Conclusions

To help ensure that a student's first hardware based DSP experience will be a positive one, software tools must be provided that are affordable, easy to install, attention getting, Windows based, and feature rich. winDSK6 meets these requirements, and with the DSK is an excellent platform to demonstrate DSP concepts in the lab and the classroom. It provides a strong audio-visual demonstration and exploration tool to help motivate students, and enhances the capabilities of an extremely powerful hardware tool.

The authors freely distribute this software for educational, non-profit use[1]. Interested parties may also contact the authors via e-mail.

[1]The winDSK6 software package is available for download at

http://www.usna.edu/EE/links/ee_links.htm

References

[1] Besterfield-Sacre, M.E., C.J. Atman, and L.J. Shuman, "How Freshman Attitudes Change in the First Year," in *Proceedings of the 1995 ASEE Annual Conferenc*e, p. 157.

[2] Landis, R.B., *Studying Engineering: A Road Map to a Rewarding Caree*r, Discover Press, Burbank, California, 1995.

[3] Ludwig, D. D., "Effects of Group Study on Student Performance in Freshman Engineering Courses," in *Proceedings of the1993 ASEE Annual Conferenc*e, p. 643.

[4] Ludwig, D.D., "An Overview of Freshman Enrollment Data and Departmental Admission Practices in Engineering Colleges," in *Proceedings of the 1995 ASEE Annual Conferenc*e, p. 228.

[5] Texas Instruments Inc., *TMS320C6711 Digital Signal Processing Starter Kit*

[6] Karplus, K., and Strong, A., "Digital Synthesis of Plucked-String and Drum Timbres," *Computer Music Journa*l, Vol. 7, No. 2, Summer 1983

MICHAEL G. MORROW, P.E., is an Associate Faculty Associate in the Department of Electrical and Computer Engineering at the University of Wisconsin - Madison. From 1997–2000 he was a Master Instructor in the Department of Electrical Engineering at the U.S. Naval Academy. His research interests include real-time digital systems, power system automation, and software engineering. He is a member of ASEE and IEEE.
Email: morrow@ieee.org

THAD B. WELCH, PhD, P.E., is an Assistant Professor in the Department of Electrical Engineering at the U.S. Naval Academy. From 1994–1997 he was an Assistant Professor in the Department of Electrical Engineering at the U.S. Air Force Academy. His research interests include multicarrier communication system design and analysis, RF channel measurements, and real-time signal processing. Commander Welch is a member of ASEE and Eta Kappa Nu and a senior member of the IEEE. Email: t.b.welch@ieee.org

CAMERON H. G. WRIGHT, PhD, P.E., is an Associate Professor and Deputy Department Head of the Department of Electrical Engineering at the U.S. Air Force Academy. His research interests include signal and image processing, biomedical instrumentation, communications systems, and laser/electro-optics applications. Lt. Colonel Wright is a member of ASEE, IEEE, SPIE, NSPE, Tau Beta Pi, and Eta Kappa Nu.
Email: c.h.g.wright@ieee.org