THIS APPLICATION CAN BE DONE USING:

PRODUCTS
- Block Diagram
- RIDE
- ImPro Lab
- VIDSP Studio
- VIDSP Suite
- OORVL Design Studio

Real-time Processing with Hypersignal RIDE

Overview

Many signal processing algorithms involve working with real-world signals for such applications as digital filtering, spectral analysis, communications systems, analytical instruments, and many more. These DSP applications typically involve the conversion of a continuous analog signal into a discrete digital representation of the signal which can then be used in a number crunching algorithm to achieve some desired result. For these applications to work successfully the data processing must occur in real-time. That is, the work to be performed on the signal must extract or determine the desired results "as it happens" with no loss of signal data.

Traditional approaches to developing an algorithm to run in real-time on DSP hardware include the tedious task of writing source code to implement the design algorithm in either C or assembly language. Using an assembler/linker/C-compiler program the source code is compiled into a form which can be implemented on a DSP device. This DSP executable file is downloaded to DSP hardware where it implements the design algorithm. This approach to developing a real-time algorithm is one which requires a great deal of programming expertise, is very time consuming and labor intensive, and detracts from the process of developing the actual design algorithm to be implemented.

This application note describes how Hypersignal RIDE can be used to quickly develop applications which require real-time processing. Using RIDE's graphical approach to implementing a real-time DSP algorithm provides a dramatic improvement over traditional design methods, and yields many benefits such as increased productivity, greater design flexibility, and vastly reduced design schedules.

Product Specific Information

Hypersignal RIDE is a superset of Hyperception's Hypersignal Block Diagram product, and as such, contains all of its simulation capabilities and functionality. However, RIDE additionally provides the capability of creating real-time DSP applications by simply connecting block function icons together with a mouse. RIDE provides full-featued COFF support, heterogeneous multiple processor support, full target DSP memory map control, interrupt hooking, DSP memory operations, symbol table, code profiling capability, multi-rate support, virtual DSP support, complete system statistics, and complete real-time DSP application export. Hypersignal RIDE also includes an extensive real-time and simulation block function library. Custom block functions are easily added through use of the included Block Wizard utility.

Hypersignal RIDE directly supports a wide variety of DSPs and DSP/acquisition hardware, and can also be used to target custom DSP hardware systems. A DATEL PCI-431 high-speed data acquisition board with up to 10 MHz A/D sampling is used in this application note.

Detailed Description

This application note will demonstrate how Hypersignal RIDE software can be used to perform data acquisition,

9550 Skillman LB125 * Dallas, TX 75243 * Tel: (214) 343-8525 * Fax: (214) 343-2457 * www.hyperception.com * E-mail: info@hyperception.com

implement digital filtering, and perform spectral analysis on a real-world signal. In this example a DATEL PCI-431 high-speed data acquisition board with an onboard TMS320C44 DSP is used in conjunction with the Hypersignal RIDE software to create a real-time DSP algorithm. It is important to mention that Hypersignal RIDE contains hundreds of functions and that many different real-time algorithms can easily be implemented using the approach described in this application note.

The example described here involves the acquisition of a real-world signal. An external signal is applied to the input channels of the DSP hardware. The hardware is programmed to acquire the signal at a specific sample rate. In addition to acquiring the signal, the data is filtered with an elliptic Infinite Impulse Response (IIR) digital filter. Both the original and the filtered signal are then transformed into the frequency domain through use of a Fast Fourier Transform algorithm. The linear magnitude of each is then calculated and the results are graphically displayed as both a 3D Waterfall and as a representation of time and frequency in a Spectrograph contour display. If desired, log scaling can be selected for both display types.

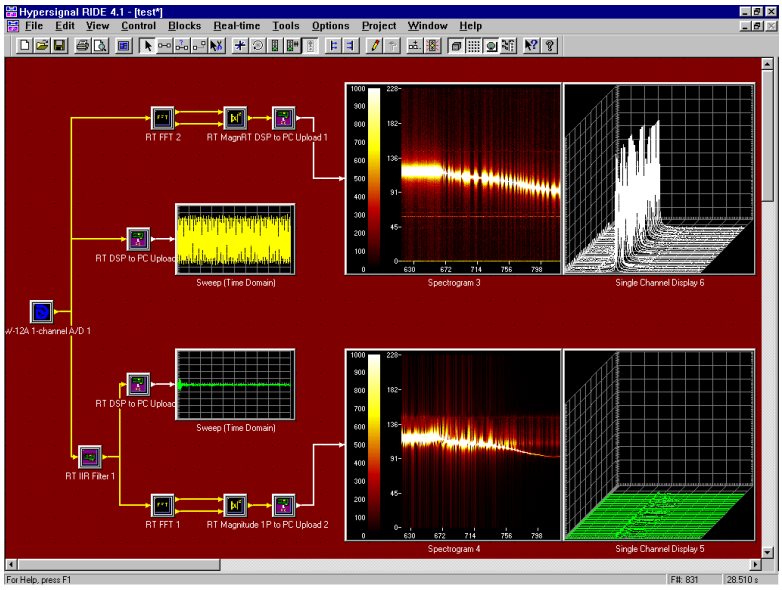

Figure 1. Implementation of a real-time algorithm in Hypersignal RIDE

Implementation

Data Acquisition

Data acquisition is a term used to describe the process of "sampling" a signal from a real-world system. In many cases, the signal captured represents the physical quantity of electric voltage transformed via the use of transducers such as microphones, thermistors, etc. In the field of digital signal processing, the transformation from a continuous analog signal to a discrete sequence is usually referred to as analog-to-digital conversion (A/D conversion).

9550 Skillman LB125 * Dallas, TX 75243 * Tel: (214) 343-8525 * Fax: (214) 343-2457 * www.hyperception.com * E-mail: info@hyperception.com

In this Hypersignal RIDE application, as shown in Figure 1, data acquisition is represented by a single block function icon located at the left-most section of the worksheet. This block represents the A/D conversion process and actually programs the DSP/acquisition hardware to sample the incoming signal. For our example, we have setup the PCI-431 board to sample at a rate of 100 kHz, and to operate on data in frames of 512 samples.

Setting A/D-specific parameters such as analog conversion sample rate, input channel selection, and data buffer framesize is easily accomplished by using the mouse to double-click the A/D block function icon and select the desired parameters. All DSP programming is handled automatically by Hypersignal RIDE and operation is transparent to the user. Data flow for the algorithm is easily established by connecting block function icons together with the mouse. Data synchronization input/output logic flags are selected at the block level to govern subsequent data processing.

For our example we have used an external waveform function generator to apply a sine wave voltage to the PCI-431's channel 0 input connection. Since we have programmed the PCI-431 hardware with a 100 kHz sample rate, the input signal used is bandlimited to 50 kHz, and the input sine frequency is varied between DC and 50 kHz.

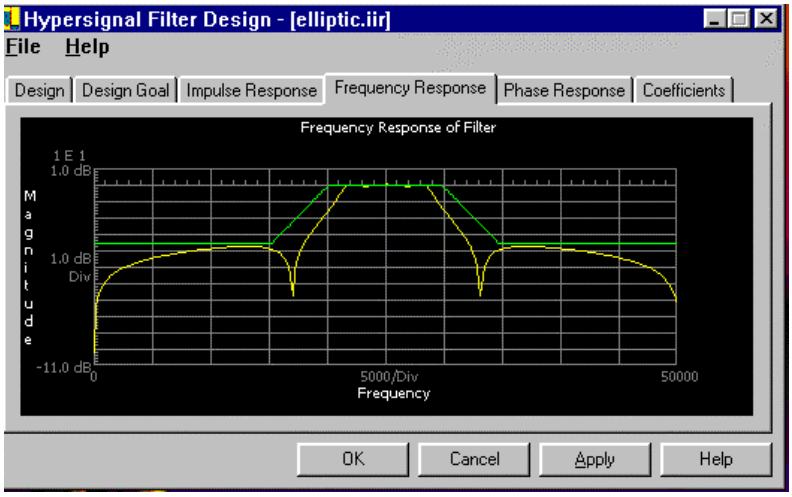

Figure 2 - IIR Coefficient Filter Design Utility

Filter Implementation

Digital filters are often characterized and designed on the basis of a desired frequency response for a given application or system. Once the appropriate filter coefficients that are required to implement the specific response have been created, they can be applied to a signal to achieve the desired results.

Hypersignal RIDE includes a Filter Design Utility, shown in Figure 2, which allows you to easily design your own custom digital filters for use in a real-time RIDE worksheet. The Filter Design Utility produces a filter coefficient

9550 Skillman LB125 * Dallas, TX 75243 * Tel: (214) 343-8525 * Fax: (214) 343-2457 * www.hyperception.com * E-mail: info@hyperception.com

file of second-order cascade sections which can be used by RIDE's dedicated real-time IIR filter block function. The filter coefficients produced by the Filter Design Utility and used by the real-time IIR block function correspond to the transfer function shown in Figure 3.

For this example a bandpass elliptic IIR filter was constructed with a passband region ranging between 20 kHz and 30 kHz. The filter's stopband attenuation was selected to be 35 dB. The general approach taken by this algorithm is to simply attenuate the input signal's frequency components which lie outside of the passband region as a means of demonstrating the filter's frequency response. As can be seen in Figure 1, the data flow of the algorithm is established with line connections. Both the input signal and the filtered waveform are uploaded for display comparison. It can be seen that the signal is heavily attenuated outside of the passband region.

$$H(z) = C \frac{\prod A_{0j} + A_{1j}z^{-1} + A_{2j}z^{-2} + \ldots + A_{mj}z^{-m}}{\prod 1 + B_{1j}z^{-1} + B_{2j}z^{-2} + \ldots + B_{nj}z^{-n}}$$

Figure 3. Transfer Function for IIR Coefficient Filter File

Transferring Time Data to Frequency Data

Often it is important for an algorithm to be able to work with the frequency components of a signal. The fast Fourier transform (FFT) algorithm is used to mathematically transform an input time-domain signal into its frequency-domain equivalent. The example found in this application note makes use of the FFT algorithm to map both the original input signal and the filtered signal into the frequency domain.

Two real-time FFT block functions are included in this example worksheet, as can be seen in Figure 1. These block functions have been setup to provide 9th-order (512 point) FFTs since the A/D input frame of data has been selected to operate with data buffers of 512 samples. The output of the FFT is complex. This is reflected in the real-time FFT's block function icon which provides two output channels - a real component and an imaginary component. The complex output from the FFT is passed on to a real-time Magnitude block function which calculates the linear magnitude of the signal.

Symbolic Information, Profiling Capability, & DSP Memory Operations

Hypersignal RIDE contains a full DSP COFF linker and provides information such as symbolic information, target DSP memory map control, complete system statistics, code profiling capability, initialization and interrupt support, and more. Figure 4, below, shows the Symbol Table for this RIDE real-time algorithm.

9550 Skillman LB125 * Dallas, TX 75243 * Tel: (214) 343-8525 * Fax: (214) 343-2457 * www.hyperception.com * E-mail: info@hyperception.com

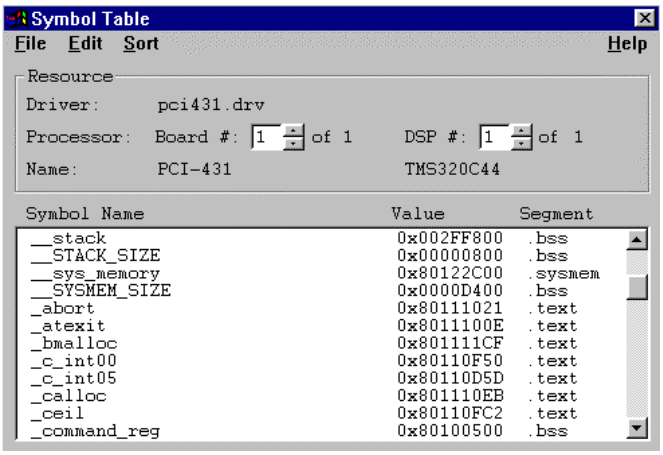

Figure 4 – RIDE Symbol Table

Spectral Comparison

It is important to keep in mind that the block diagram represented in the worksheet is actually being executed on the target DSP hardware, and not on the PC. All computational results are resident in DSP memory space. Hypersignal RIDE provides full communication between the DSP & the PC and allows you to easily read and write DSP memory. The results from this example are uploaded from the DSP hardware via DSP2PC Upload block functions and the resulting data is displayed graphically. As can be seen in the display graphs in Figure 5, below, the original signal can be compared to the bandpass filtered version.

It is readily apparent from both the 3D Waterfall and Spectrograph contour displays that the signal has been attenuated outside of the passband region of 20 kHz to 30 kHz.

9550 Skillman LB125 * Dallas, TX 75243 * Tel: (214) 343-8525 * Fax: (214) 343-2457 * www.hyperception.com * E-mail: info@hyperception.com

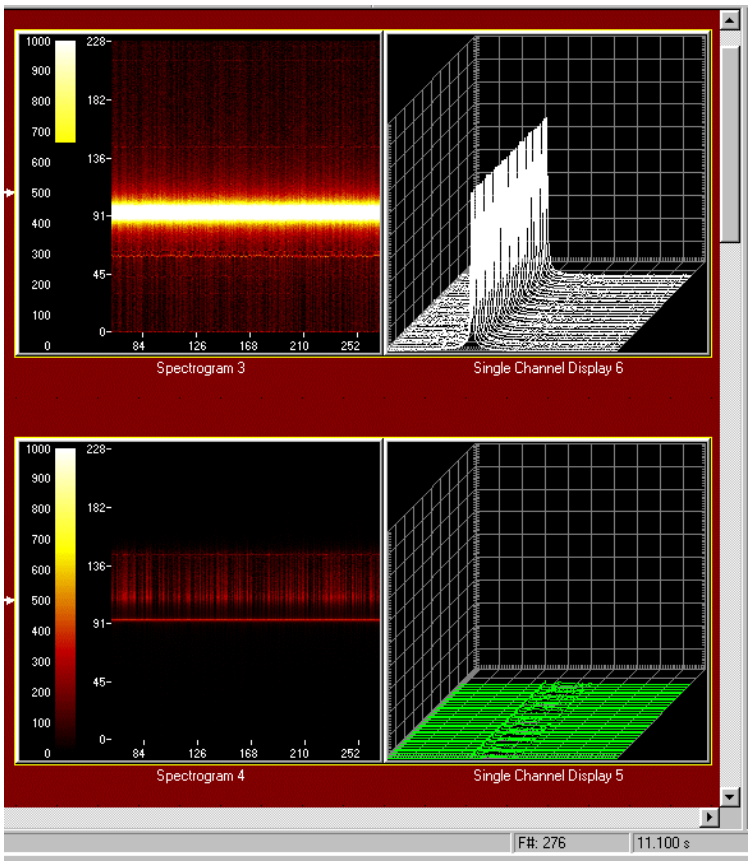

Figure 5 – Spectral Comparison of Filtered and Original Signal

Applications

The implementation of a real-time DSP application is quickly achieved with Hypersignal RIDE software. By simply creating a block diagram representation of the desired algorithm via block function icons and line connections the user can efficiently create and test an algorithm on a target DSP. The resulting code can also be exported to a standard COFF file for use in embedded DSP applications.

The entire design has been conducted in the Hypersignal RIDE environment without the need for writing and debugging time-consuming code which is required by traditional methods.

Hyperception
The Leader in DSP

9550 Skillman LB125
Dallas, Texas 75243
Tel: (214) 343-8525 * Fax: (214) 343-2457
E-Mail: info@hyperception.com
www.hyperception.com