THIS APPLICATION CAN BE DONE USING:

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

DTMF Generation with Hypersignal Block Diagram

Overview

Dual-Tone Multi-Frequency (DTMF) is the generic name for pushbutton telephone signaling equivalent to the Bell System's TouchTone. DTMF signaling is quickly replacing dial-pulse signaling in telephone networks worldwide. In addition to telephone call signaling, DTMF is becoming popular in interactive control applications, such as telephone banking or electronic mail systems, in which the user can select options from a menu by sending DTMF signals from a telephone.

This application note describes a DTMF coding and decoding implementation based on Hypersignal Block Diagram. The DTMF coding is based on two tones used to generate a digit. Two of eight tones can be combined so as to generate sixteen different DTMF digits. The DTMF decoding is based on the discrete Fourier transform (DFT). Using this algorithm, 16 DTMF receivers can be implemented on 4 four frequency Goertzel algorithm function blocks . Then, the output of Goertzel algorithm can be decoded into a digit.

Product Specific Information

This application could be designed using any edition of Hypersignal Block Diagram or Hypersignal RIDE (Standard Edition, Professional Edition, or Enterprise Edition). In this example, no DSP/Acquisition board is needed, as the PC is doing all the processing; however, it would be relatively easy to implement this in real-time using a DSP/Acquisition board in conjunction with Hypersignal RIDE.

Detailed Description

In a DTMF signal generation, a DTMF keypad could be used for digit entry, the resultant DTMF tones are generated mathematically and added together. The values are logarithmically compressed and passed to the receiver. In a DTMF scheme, pairs of tones are used to signal the digits 0 through 9, pound(#), star(*), and the digits A, B, C and D. For each pair, one of the tones is selected from a low group of four frequencies, and the other from a high group of four frequencies. The correct detection of a digit requires both a valid tone pair and the correct timing intervals.

The matrix of frequencies used to encode the 16 DTMF symbols is shown in Figure 1. Each symbol is represented by the sum of the two frequencies that intersect the digit. The row frequencies are in a low band, below 1 kHz, and the column frequencies are in a high band, between 1 kHz and 2 kHz. The digits are displayed as they would appear on a telephone's 4x4 matrix keypad (on standard telephone sets, the fourth column is omitted). The user should note that there are a number of different algorithms possible for generation and detection of DTMF tones; this application note simply describes one manner of doing so. Also, due to the fact that Hypersignal Block Diagram/RIDE is ever evolving, certain block functions may be replaced by newer functions, and some blocks may even become obsolete. Therefore, this application note should be considered simply a guide, to be used for whatever insights it gains the user towards their specific application.

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

	column 1 1209 Hz	column 2 1336 Hz	column 3 1477 Hz	column 4 1633 Hz
row 1 697 Hz	1	2	3	A
row 2 770 Hz	4	5	6	B
row 3 852 Hz	7	8	9	C
row 4 941 Hz	*	0	#	D

Figure 1. DTMF Digits

Implementation

DTMF signal generation

In the Hypersignal Block Diagram application, as shown in Figure 2, DTMF signal generation can be implemented relatively straightforward. The sample rate for the system is set using the global parameters, Sample Rate, and Framesize. For this application the system sample rate is set to be 11.025 kHz (mainly due to the use of a PC-based sound card which will be driven real-time from this application; generally, 8 kHz would be the choice for most telecom applications). The framesize was arbitrarily selected to be 1000 samples.

In this example, the two Sine Generators are used to generate the row and column tones which are then added together to form the DTMF. The frequencies of the two generators are controlled through the keypad, which generates the selected row and column which are used by the row and column table lookup block functions. The output of the table lookup blocks correspond to frequencies which match those in Figure 1. Using the Parameter Connect mode, the outputs of the Table Lookup blocks are connected to the Frequency parameter of each of the sine generators, allowing for a dynamic frequency control while running.

The selected tones are added together, the resultant waveform is sent to a gain block, (the level of which is controlled by a knob and gated with a keypad 'button pressed' value) and sent to a display for viewing. In order to allow the DTMF to be heard by the user, the generated signal is also sent to a sound card, showing an example of how fast Hypersignal Block Diagram is; the tone is actually generated/heard in real-time!

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

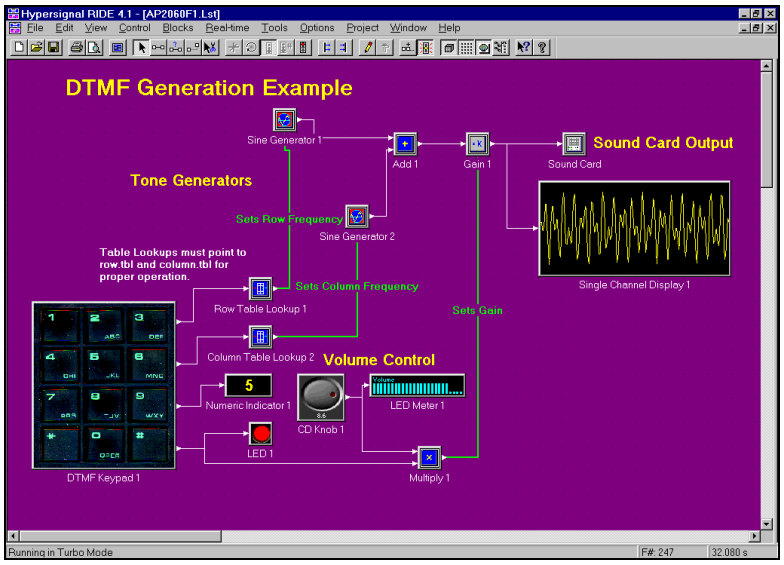

Figure 2 - Implementation of a DTMF Coding in Hypersignal Block Diagram

Decoding DTMF Signals

Decoding a DTMF signal involves extracting the two tones in the signal and determining from their value the intended DTMF digit. Tone detection is often done in analog circuits by detecting and counting zero-crossings of the input signal. In digit circuits, tone detection is easier to accomplish by mathematically transforming the input time-domain signal into its frequency-domain equivalent by means of the Fourier transform, or through use of tone-specific digital filters.

The general approach taken by this algorithm for DTMF tone detection is to take the Fourier transform of the observed signal and search for energy at the frequencies of interest. Since the algorithm is implemented by Discrete Fourier Transform (DFT). The analysis frame must be long enough to resolve the DTMF frequencies, but short enough to detect the minimum length tone. A 12.75 ms frame at a sampling rate of 8kHz is good choice.

In calculating the DFT, the Goertzel algorithm, a method for calculating any single coefficient of a DFT, is chosen over a fast Fourier transform (FFT) algorithm. There are two reasons for this. In order to obtain the required frequency resolution at an 8 kHz sampling rate, a 256-point FFT would be required. Since the algorithm for tone detection requires knowledge of the energy at only 16 frequencies, it is more efficient to execute the Goertzel algorithm for these frequencies. In addition, the Goertzel algorithm is recursive, eliminating the need to store 256 samples for the FFT for each DTMF dector. This saves both time and data memory in the simulation and the real time applications.

The Goertzel algorithm can be thought of as a second-order IIR filter. for each DFT coefficient. The transfer function for the filter is:

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

$$H_k(Z) = (1-W_N^kZ^{-1})/(1-2\cos(2\pi k/N)Z^{-1}+Z^{-2})$$

where $W_N = \exp(-j2\pi/N)$, and N is the length of the observation window for the DFT. The flow chart of this transfer function is shown in Figure 3.

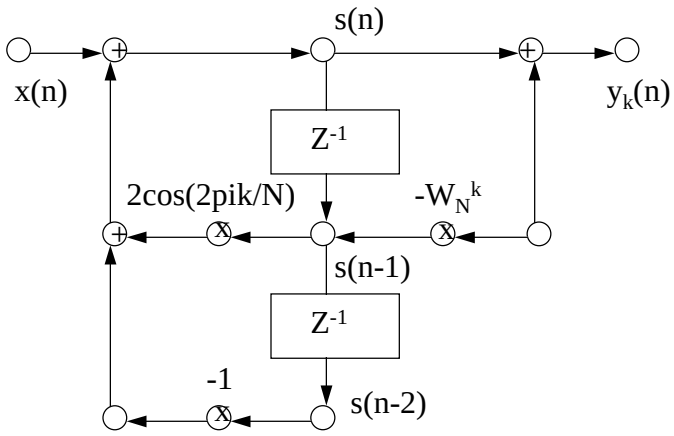

Figure 3 - Flow Chart of the Transfer Function

Initially, the state variables of the filter are set to zero. Then, the filter is executed N times. The output at this point, $y_k(N)$, is the k'th coefficient of a length N DFT. Notice that the filter is implemented as a direct form II second-order section. The recursive part of the filter is on the left-hand side of the delay elements, and the nonrecursive part is on the right. Since only the output at time N is needed, it is only necessary to compute the nonrecursive part of the filter after the last iteration of the recursive part. A further simplification in the algorithm is made by observing that only the square of the magnitude of the DFT coefficient is needed. The nonrecursive calculation of the DFT coefficient is:

$$y_k(N) = s(N) - W_N^k s(N-1)$$

where s(N) and s(N-1) represent the value of the state variables at times N and N-1. It can be shown that:

$$|y_k(N)|^2 = |s(N)|^2 - 2\cos(2\pi k/N)s(N)s(N-1) + |s(N-1)|^2$$

Therefore, it is only necessary to store the value, $2\cos(2\pi k/N)$, for each coefficient to be evaluated.

The implementation of a DTMF decoding in Hypersignal Block Diagram is shown in Figure 4. This application makes use of a simple DTMF Test Tone Generator, made up of two Sine Generator blocks and an Add block. DTMF decoding is done by finding the maximum row and maximum column energy. This task is performed by using 2 Four Frequency Goertzel Algorithm blocks. In each four freq. Goertzel Algorithm block, the magnitude

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

of four frequencies can be detected simultaneously. These four magnitudes are then used by the 4-input Max function to determine the zero-based index of the winning frequency. One of the Four Frequency Goertzel Algorithm/4-input Max pair of blocks is used to determine the row index, while the other is used to determine the column index. Then the detected row index is shifted left by 2 and logically OR'ed with the column index to create a zero-based index representing the detected DTMF. Although not implemented in this example, a level detect could be implemented to discriminate against noise; in this example, the detection is always running, providing a 'best-guess' DTMF digit even while no valid DTMF's are presented. One way to accomplish this would be to use the other output of the 4-input Max blocks, which are delivering the magnitudes of the detected row and column, in conjunction with a minimum energy detection algorihtm, comprised mainly of two threshold blocks.

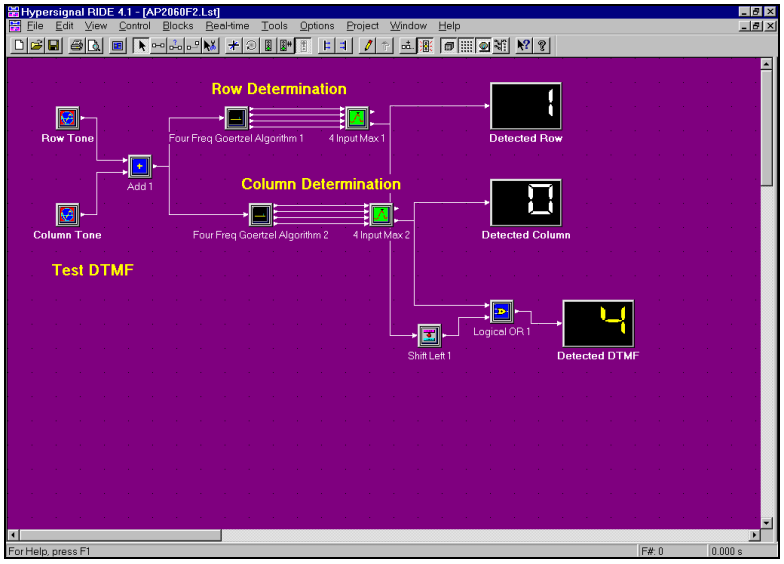

Figure 4. The Implementation of DTMF Decoding in Hypersignal Block Diagram

DTMF Coding and Decoding Simulation

The application of DTMF coding and decoding in Hypersignal Block diagram is shown in Figure 5. As used earlier, the DTMF Keypad block allows digit entry and has four outputs. The first output is the row number of the key that was last pressed (0-3), the second output is the column number of the key that was last pressed (0-3), the third output is the value of the key pressed, and the fourth output is 1 if a key is being pressed, 0 otherwise. The generation of DTMF is performed as in the intial example, and the decoder example has been added, with the Test Tone Generator replaced with a sound card A/D to actually get real-time signals. Assuming a standard PC-based sound card, with proper connections to a microphone and speakers which are physically close to each other, pressing the DTMF keypad will result in a detected zero-based index.

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

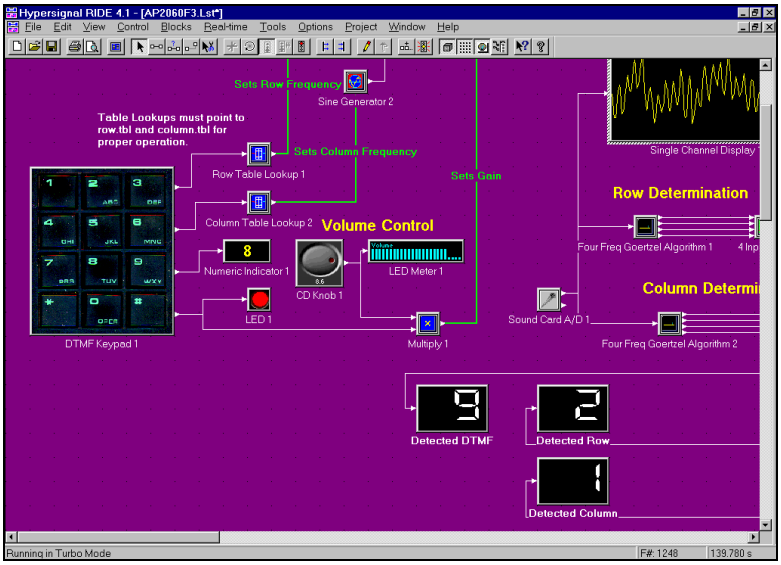

Figure 5 - The Implementation of DTMF coding and Decoding in Hypersignal Block Diagram

Applications

The implementation of the DTMF coding and decoding is one of the examples of the power and flexibility of Hypersignal Block Diagram. The successful simulation design can be generated into ANSI C source code by using Hypersignal C source code generator, included in the Enterprise Edition of Hypersignal Block Diagram/RIDE. The resultant C source code can be cross-complied into the different DSP code and implemented in real time DSP hardware. The entire design has been conducted in the Hypersignal Block Diagram without the need for writing and debugging time-consuming simulation code required by the traditional methods.

References

1. Amy Mar, Digital Signal Processing Applications Using the ADSP-2100 Family, PRENTICE HALL,1990

2. Kun-Shan Lin, Digital Signal Processing Applications with TMS320 Family Theory, Algorithms, and Implementations, Texas Instruments, 1990

3. J. Hartung, S.L. Gay, and G.L. Smith, Dual-Tone Multifrequency Receiver Using the WE DSP32 Digital Signal Processor, AT&T Application Note, June 1988

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