Volume : 1 | Issue : 6 | Nov 2012 • ISSN No 2277 - 8179

Research Paper

Engineering

KEYWORDS : DTMF detection, Goertzel, resource sharing.

Design and Implementation of Goertzel Algorithm for DTMF application on FPGA

V.Siva — GURU NANAK INSTITUTE OF TECHNOLOGY, IBRAHIMPATNAM, HYDERABAD, ANDHRA PRADESH

shaik. saidulu — GURU NANAK INSTITUTE OF TECHNOLOGY, IBRAHIMPATNAM, HYDERABAD, ANDHRA PRADESH

ABSTRACT

Dual-tone multi-frequency (DTMF) is a signaling standard in telecom applications that produces two tones simultaneously for each key press. The DTMF tone detection is very crucial block in several telecom based embedded systems. As the current generation embedded systems are looking for key feature of low power, the DTMF detection algorithm also must be implemented with low power schemes. The DTMF detection can be done with FFT based technique which is power consuming type and it requires more hardware. The modified Goertzel algorithm can detect the incoming frequency within a ±1.5% offset range. This algorithm does not check for overflow problems, nor is it a complete detection algorithm. To ensure complete detection, further evaluation of the detected tone in the form of many tests is required. These tests could include twist tests, dynamic tests, guard time tests, signal-to-noise ratio tests, and talk off tests.

I.INTRODUCTION

DTMF signaling has many applications such as telephone dialing, data entry, credit checking, and voice mail system control. A DTMF signal consists of two superimposed sinusoidal waveforms with frequencies chosen from a set of eight standardized frequencies. These frequencies should be generated and detected according to the CCITT Recommendation

II.BLOCK DIAGRAM:

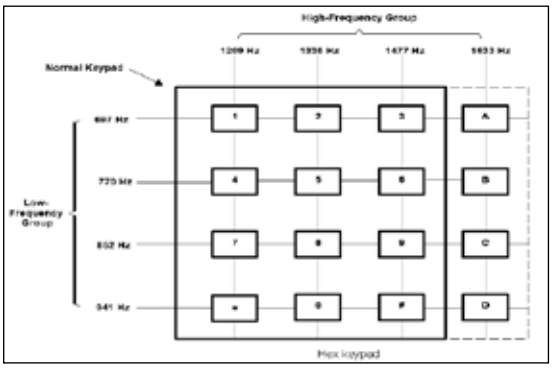
General module

The module DTMF detection consists of

- Hex key pad
- DTMF test signal generator
- Additive white Gaussian noise
- Frequency Detection block
- Magnitude / index estimator
- Frequency to digit look-up table

1. Hex keypad

Hex Keypad gives input to the module. It is an external component. The signal from column is taken as input. Row and display are the output signals.



2. Frequency test signal generator:

Block generates the carrier frequencies necessary. It consists of two blocks

a. Frequency word selector:

In this block the carrier waves are generated according to the key pressed o

For example: If key 9 is being press the frequencies that are generated are 852 Hz (Low frequency group) and 1477 Hz (High frequency group These frequency waves are generated by a DDS core.

b. DDS Core:

The Logic ORE™ IP DDS (Direct Digital Synthesizer) Compiler core sources sinusoidal waveforms for use in many applications.

A DDS consists of a Phase Generator and a SIN/COS Lookup Table. These parts are available individually or combined via this core. Direct digital synthesis (DDS) is a method of producing an analog waveform—usually a sine wave—by generating a time-varying signal in digital form and then performing a digital-to-analog conversion

Modulation controller:

The modulation controller loads phase increment value with one (p=1), to generates output frequency corresponding to input symbol '0' and with two (p=2), which is to generate output frequency corresponding to input symbol '1'. This phase increment value is given by frequency word selector.

The value of phase increment is calculated by the formula: ΔP= (fs * 2j)/ fclkThe phase increment value for the frequencies is as follows:

Clock distributor and control signal generator

The phase accumulator has to reset (phase register becomes zero) whenever a new symbol is loaded by modulation controller. This is achieved by resetting the phase accumulator.

The clock input for phase accumulator has to be derived from the master clock so that the output wave form gets an integer number of cycles for every symbol. The Clock distributor and control signal generator blocks generates necessary signals for above two requirements

Phase Accumulator:

Digital phase accumulator gives the instantaneous phase of a cosine signal according to a phase increment value (frequency word) given as input to accumulator by BFSK modulation controller.

When the accumulator exceeds a value equivalent to 2pi radians, it overflows, multiples of 2pi are discarded, and the incrementing process continues to the next cycle.

The output of phase accumulator will produce synthesized instantaneous digital phase, which will be used to address the LUT.

COS carrier LUT:

The phase bits given by the accumulator is used to address a look-up table held in ROM (read- only memory) which gives corresponding amplitude bits. The COS carrier LUT consists of phase Vs amplitude table corresponding to one COS waveform.

The main propose of this block is take two cosine waves from DDS cores and add them in order to produce one wave called tone out.

3. Additive white Gaussian Noise:

The tone out, which is the output from tones generator is mixed

Volume : 1 | Issue : 6 | Nov 2012 • ISSN No 2277 - 8179

with noise in this module. The output is named as noise bits.

Wideband Gaussian noise comes from many natural sources, such as the thermal vibrations of atoms in conductors (referred to as thermal noise or Johnson-Nyquist noise), shot noise, black body radiation from the earth and other warm objects, and from celestial sources such as the Sun.

4. Frequency Detector:
Input to this module is noise bits, which is the output from additive white Gaussian noise.

Output of the block is indices and magnitudes. As per the scope of the project, there are three variants of frequency detector block

1. FFT-128 core
2. Goertzel algorithm
3. Resource sharing

1. FFT- 128 core:
The Xilinx Logic ORE™ IP Fast Fourier Transform (FFT) implements the Cooley-Turkey FFT algorithm, a computationally efficient method for calculating the Discrete Fourier Transform (DFT). A fast Fourier transform (FFT) is an efficient algorithm to compute the discrete Fourier transform (DFT) and it's inverse. There are many distinct FFT algorithms involving a wide range of mathematics, from simple complex-number arithmetic to group theory and number theory.

A DFT decomposes a sequence of values into components of different frequencies. This operation is useful in many fields (see discrete Fourier transform for properties and applications of the transform) but computing it directly from the definition is often too slow to be practical. An FFT is a way to compute the same result more quickly: computing a DFT of N points in the naive way, using the definition, takes O(N2) arithmetical operations, while an FFT can compute the same result in only O(N log N) operations. This huge improvement made many DFT-based algorithms practical; FFTs are of great importance to a wide variety of applications, from digital signal processing and solving partial differential equations to algorithms for quick multiplication of large integers.

The most well known FFT algorithms depend upon the factorization of N, but (contrary to popular misconception) there are FFTs with O (N log N) complexity for all N, even for prime N. Many FFT algorithms only depend on the fact that $e^{-\frac{2\pi i}{N}}$ is an N primitive root of unity, and thus can be applied to analogous transforms over any finite field, such as number-theoretic transforms.

Overview :The FFT core computes an N-point forward DFT or inverse DFT (IDFT) where N can be 2m, m = 3–16. For fixed-point inputs, the input data is a vector of N complex values represented as dual b x -bit two's-complement numbers, that is, b x bits for each of the real and imaginary components of the data sample, where b x is in the range 8 to 34 bits inclusive. Similarly, the phase factors b w can be 8 to 34 bits wide.

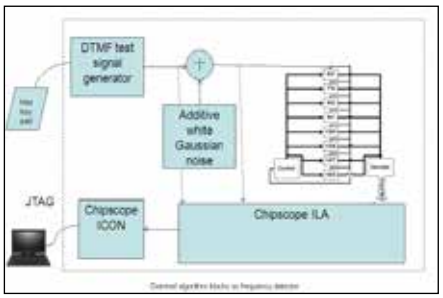
The point size N, the choice of forward or inverse trans-form, the scaling schedule and the cyclic prefix length are run-time configurable. Transform type (forward or inverse), scaling schedule and cyclic prefix length can be changed on a frame-by-frame basis. Changing the point size resets the core. Four architecture options are available: Pipelined, Streaming I/O, Radix-4, Burst I/O, Radix-2, Burst I/O, and Radix-2 Lite, Burst I/O.

Here N is the transform size and. $j = \sqrt{-1}$ erse DFT (IDFT) is given by

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{jnk2\pi/N} \quad n = 0,\ldots,N-1$$

Algorithm:
The FFT core uses the Radix-4 and Radix-2 decompositions for computing the DFT. For Burst I/O architectures, the decimation-in-time (DIT) method is used, while the decimation-in-frequency (DIF) method is used for the Pipelined, Streaming I/O architecture. When using Radix-4

2. GOERTZEL ALGORITHM:



The Goertzel algorithm is a digital signal processing (DSP) technique for identifying frequency components of a signal, published by Dr. Gerald Goertzel in 1958. While the general Fast Fourier transform (FFT) algorithm computes evenly across the bandwidth of the incoming signal, the Goertzel algorithm looks at specific, predetermined frequencies. Some applications require only a few DFT frequencies.

One example is frequency-shift keying (FSK) demodulation, in which typically two frequencies are used to transmit binary data; another example is DTMF, or touch-tone telephone dialing, in which a detection circuit must constantly monitor the line for two simultaneous frequencies indicating that a telephone button is depressed.

Goertzel's algorithm reduces the number of real-valued multiplications by almost a factor of two relative to direct computation via the DFT equation.

For a length of N, the Goertzel's series is

$$H_k(z) = \frac{1 - W_N^k z^{-1}}{1 - 2\cos(\frac{2\pi k}{N})z^{-1} + z^{-2}}$$

Where,

$$k = 0,1,\ldots, N-1$$

3. RESOURCE SHARING:
Resource sharing is the assignment of similar block of operations (for example, +) to a common net list cell. Net list cells are the resources—they are equivalent to built hardware. Resource sharing reduces the amount of hardware needed to implement block operations. Without resource sharing, each operation is built with separate circuitry.

Input signal x is coming at the chosen sample rate (in our design it is 5 KHz). The RS based GA shall work at 8 times higher frequency (with rising edge coincidence).

In this approach instead of a simple register (Z-1 in the above diagram) a RS buffer will be used.

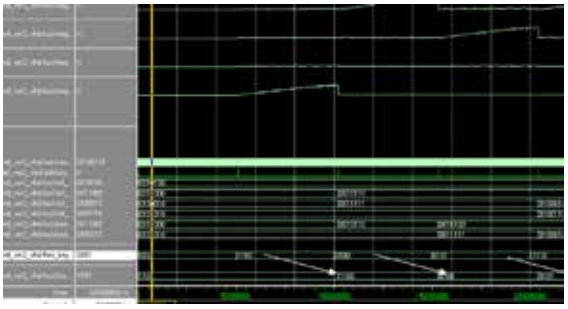
III. IMPLEMENTATION
Since the telephone industry has preset the sampling frequency to 8 kHz and the DTMF frequencies to 697,770, 852, 941, 1209, 1336, 1477, and 1633 Hz, the filter length must be large enough to find the desired K value that corresponds to the DTMF frequencies. Therefore, there is a trade off to be considered between the computation burden and better resolution. For this application report, the filter length, N, was chosen as 105, which is the smallest value that can fulfill DTMF detection.(3) Table 1 shows the calculated k values for N = 105.

The following specifications are considered for design and implementation of the DDFS.

1) Types of signals: sine.
2) Type of signal synthesis : ROM based DDFS
3) phase resolution chosen in DDFS :5.625O

4) Front end design entry: VHDL
5) Backend synthesis : Xilinx Spartan 3E FPGA

Output wave form of hex key values found



IV. CONCLUSIONS

The modified Goertzel algorithm can detect the incoming frequency within a ±1.5% offset range. This algorithm does not check for overflow problems, nor is it a complete detection algorithm. To ensure complete detection, further evaluation of the detected tone in the form of many tests is required. These tests could include twist tests, dynamic tests, guard time tests, signal-to-noise ratio tests, and talk off tests.

ACKNOWLEDGMENT

I would like to articulate my profound gratitude and indebtedness to Dr. T Srinivasulu, Principal, and GNIT. And Prof. B. Kedarnadh, Head of the Department, GNIT, for guiding and encouraging me in all aspects.

I wish to extend my sincere thanks to all the faculty members of Guru Nanak Institute of Technology.

REFERENCE

1. C. Mervin, "General-Purpose Tone Decoding and DTMF Detection," in Theory, Algorithms, And Implementations, Digital Signal Processing Applications with the TMS320 Family, Vol. 2, Literature number SPRA016, Texas Instruments (1990). | 2. J. G. Proakis and D. G. Melonakos, Digital Signal Processing: Principles, Algorithms, and Applications, 2nd ed., Macmillan, New York, NY (1992). | 3. G. L. Smith, Dual-Tone Multi frequency Receiver Using the WE DSP16 Digital Signal Processor, AT&T Application Note. | 4. P. Mock, "Add DTMF Generation and Decoding to DSP-mP Designs," in Theory, Algorithms, And Implementations, Digital Signal Processing Applications with the TMS320 Family, Vol. 1, Literature number SPRA012A, Texas Instruments (1989). | 5. Y. C. Huang, "DSP Techniques for Telecommunication Systems," Master's thesis, Northern Illinois University (August 1995).