Efficient Implementation of Transform Based Audio Coders using SIMD Paradigm and Multifunction Computations

Luckose Poondikulam S (luckose@sasken.com), Suyog Moogi (suyog@sasken.com),
Rahul Kumar, K P P Kalyan Chakravarthy, Vaibhav Mathur, Barker Nalanikuzhi B
Sasken Communication Technologies Limited, Bangalore, India.

Abstract

This paper illustrates the suitability of ADSP 21160 for implementation of high performance DSP applications requiring 32 bit floating point precision. We present a real-time implementation of a multi-channel MPEG-2 AAC-LC encoder. The performance results of this implementation are also discussed in this paper.

Keywords MPEG-2, AAC, Multi-channel audio, SHARC, ADSP 21160, SIMD

1 Introduction

The Analog Devices' SHARC family architecture [1] is designed with an emphasis on sustained signal processing performance. The newest member of the SHARC family is the ADSP-21160 [2]. This new DSP, which is code compatible with the popular ADSP-2106x family, offers a SIMD floating-point processor core. The core's high mathematical performance is balanced with a large internal dual-ported SRAM, zero-overhead DMA and integrated multi-processing features. Like all members of the SHARC family, the ADSP-21160 processes 32-bit fixed-point, 32-bit IEEE floating-point, and 40-bit floating-point data types.

The standardization body, Motion Picture Experts Group (MPEG) in its first phase of specifications came out with MPEG-1 audio coding system [4], which operates in single-channel or two-channel stereo modes. Later MPEG defined a multi-channel extension to MPEG-1 audio standard which is backward compatible (BC) with MPEG-1 systems. MPEG-2 BC [5] provides good quality audio at data rates of 640-896 kbps for five full-bandwidth channels.

The MPEG-2 advanced audio coding (AAC) system was designed to provide the best audio quality for multi-channel case without any restrictions due to compatibility requirements. MPEG-2 AAC combines the coding efficiency of a high resolution filter-bank, prediction techniques and Huffman coding with additional functionality aimed to deliver high audio quality at a variety of data rates. MPEG-2 AAC satisfies the ITU-R quality requirements [7] at 320 kbps per five full bandwidth channels. The MPEG-2 AAC scheme is also the kernel of the MPEG-4 audio standard.

In this paper, an implementation of a real time MPEG-2 AAC Low Complexity (LC) Profile encoder for multi-channel (up to 5.1 channels) audio is discussed.

2 ADSP 21160 Architecture

Some of the important factors and characteristics which decide the efficacy of any DSP are its raw processor speed, real world processor speed, suitability of processor core architecture for DSP algorithms, ease of system integration, multi-processing support, ease of coding and time to market.

The ADSP-21160 has six computation units on two data paths and runs at 100 MHz, providing 600 MFLOPS of performance. This 600 MFLOPS number is achieved in the ADSP-21160 in useful algorithms such as the FFT. Because of its

efficient architecture, the ADSP-21160 produces better algorithmic benchmarks, such at the FFT, with a slower clock rate.

The SIMD SHARC core is comprised of three main units:

- Program sequencer
- Data address generators (DAGs)
- Computational unit

The ADSP 21160 program flow and branching is controlled by the program sequencer. Data address generation and circular buffering is accomplished with the two DAGs (DAG1 and DAG2). There are 96 registers dedicated to the DAGs. The computation units, the DAGs, and the program sequencer each have their own register sets, because of which, SHARC is less susceptible to problems related to register limitations.

The computational unit is made up of two processing elements (PE) that support the SIMD SHARC core. SIMD is an acronym for single instruction multiple data. In this architecture a single instruction is issued to both processing elements. When executing this instruction each PE operates on different data. This SIMD architecture is optimal in terms of performance, code size, and power conservation for signal processing applications. Each PE in the SIMD SHARC includes

- 732-/40-bit fixed- and floating-point ALU
- 32-/40-bit floating-point multiplier with 80-bit accumulator
- Barrel shifter
- Thirty-two 40-bit data registers

This totals to 2 ALUs, 2 MACs, 2 shifters, and 2 registers files in this processor core. Each register file contains 16 primary and 16 alternate 40-bit registers. Mathematical operations are performed on data contained in the 40-bit registers in the register file. All operations in the ADSP-21160 execute in a single cycle.

With SIMD architecture, a single instruction controls multiple computational units. In a single processor cycle, executing a single 48-bit instruction, the ADSP-21160 can execute 6 floating-point operations and two pointer updates.

The ADSP-21160 has a three-stage instruction pipeline. The worst case delay for a branch is two processor cycles. Delayed branches are supported so that programmers can take advantage of these two processor cycles to execute instructions. Special logic has been added to the program sequencer to support zero overhead looping. At the end of a loop the SHARC can execute an instruction, check the loop condition, decrement the loop counter and branch, all in a single cycle. These loops are interruptible and can be nested six levels deep. The ADSP-21160 also gives the programmer the choice to make instructions conditional or non-conditional. The conditions are based on flags in registers expressly designed for this operation. The execute pipeline in the SHARC is single-cycle. All arithmetic, logic, multiplication, and load and store instructions make their results available on the cycle after issue.

In real-world DSP applications, internal memory size is critical to performance. Small internal memories usually require instructions and data to be stored off chip. Since external memory accesses can be slow, this can directly and adversely affect the performance of the processor. The ADSP-21160 has 4 Mbits of on-chip, dual-ported memory that can be freely allocated to Program Memory (PM) or Data Memory (DM). Because the memory is dual ported, the core and the DMAs can access the internal memory in the same cycle without incurring any penalty. Consequently, the SHARC has zero overhead DMAs, enabling the DMAs to run concurrently with the core. Zero overhead DMA greatly improves overall performance because the core is not burdened with the task of dealing with I/O. Further, the ADSP-21160 has superior code density and hence

will use less memory locations for instructions.

3 MPEG – 2 AAC-LC

MPEG-2 AAC [6] combines the coding efficiency of a high resolution filter-bank, prediction techniques and Huffman coding to achieve broadcast quality audio at very low bit rates. In order to allow a trade-off between the quality and the memory and the processing power requirements, the AAC system offers three profiles: Main profile, Low Complexity (LC) profile and Scalable Sampling Rate (SSR) profile. The current implementation is for LC profile. In this configuration, the following tools are used:

- Filter-bank
- Temporal Noise Shaping (TNS)
- Quantization and Coding
- Noiseless Coding
- Mid/side stereo coding (M/S)
- Bit-stream multiplexing

4 Implementation on ADSP 21160

4.1 Psycho-acoustic Model

The output from the psycho-acoustic model is:

1. A set of Signal-to-Mask Ratios (SMR) and thresholds which are adapted to the encoder.
2. The delayed time domain data (PCM samples), which are used by the MDCT
3. The block type for the MDCT (long, start, stop or short type)
4. An estimation of how many bits should be used for encoding in addition to the average available bits.

A proprietary psycho-acoustic model is used to compute SMRs, thresholds and the block type. The majority of the computations involved in this model are in FFT, convolutions, dot products etc which can be very efficiently realized on ADSP 21160. The built-in bit-reversed addressing combined with single cycle multi-function instructions in ADSP 21160 enable highly efficient implementation of FFT. Most of these computations involved, exploit SIMD, hardware loops & single cycle multifunction instructions very efficiently.

4.2 Filter-bank

A fundamental component of the MPEG-2 AAC system is the conversion of the time domain signals at the input of the encoder into an internal time-frequency representation. This conversion is done by a forward MDCT. The analytical expression for the MDCT is

$$X_{ik} = 2\sum_{n=0}^{N-1} x_{in} \cos\left[\frac{2\pi}{N}(n+n_0)\left(k+\frac{1}{2}\right)\right], \quad k = 0,\ldots,\frac{N}{2}-1$$

MDCT coefficients can be computed using any fast algorithm available. Windowing is combined into the first stage of the flow graph to exploit SIMD efficiently. The M/S stereo coding decision is taken based on the energies in the channel pairs. Apart from the MDCT coefficients, the energies in the scalefactor bands are also computed in this module. For the LFE (low frequency effects) channel, a separate scheme is used to compute the 13 coefficients [6].

4.3 Temporal Noise Shaping (TNS)

The Temporal Noise Shaping tool, provided in AAC, helps in better handling of transients and pitched input signals. Coding is difficult is these conditions because of temporal mismatch between masking threshold and quantization noise (also known as pre-echo problem). TNS is applied separately on the spectrum of every full bandwidth channel. TNS is not applied for the LFE (Low frequency effect) channel. This module involves computation of autocorrelation, filtering & finding the prediction coefficients. These compute intensive modules can be realized efficiently

using zero loop overhead and single cycle multifunction instructions in SIMD mode.

4.4 Quantization & Coding

The primary goal of this module is to quantize the spectral data in such a way that the quantization noise fulfills the demands of the psycho-acoustic model. At the same time the number of bits needed to code the quantized spectrum must be below a certain limit. AAC uses nonlinear quantization. The use of nonuniform quantizer is, of course, not sufficient to fulfill psycho-acoustic demands. An additional method for shaping the quantization noise is required. AAC uses individual amplification of groups of spectral coefficients, the so-called scale factor bands. In order to fulfill the requirements as efficiently as possible, it is desirable to be able to shape the quantization noise in units similar to the critical bands of the human auditory system. In AAC it is possible to build groups of spectral values which reflect the bandwidth of the critical bands very closely. The decision as to which scale factor band has to be amplified is, within certain limits, left up to the encoder. Two loops, the so called inner and outer iteration loops are used for determining optimum quantization. The task of the inner iteration loop is changing the quantizer step size until the given spectral data can be encoded with the available bits. The task of the outer iteration loop is amplifying the scale factor bands in such a way that the demands of the psycho-acoustic model are fulfilled as far as possible. The input to the noiseless coding module is the set of quantized spectral coefficients. Noiseless coding segments the set into sections, such that a single Huffman code book is used to code each section and this varies from block to block. In the case of short window, grouping and interleaving is used to improve the coding efficiency.

A proprietary compute inexpensive method for rate control is used to achieve good quality. A method which involves table look-up and 2 multiplications is used to compute the quantized spectrum which can be very efficiently realized on 21160. The number of iterations in each loop is also controlled to ensure real-time encoding of 5 full band width channels and the LFE channel without degrading the quality. Counting of the number of bits necessary to encode the quantized data is compute intensive and involves lot of decision making and searching for optimal codes. This step was restructured to suit 21160 architecture.

4.5 Bit-stream formatting:

Since the AAC system has a bit buffer that permits its instantaneous data rate to vary as required by the audio signal, the length of each block is not constant. In this respect the AAC bit stream uses variable-rate headers. These headers are byte-aligned so as to permit editing of bit streams at any block boundary.

The instructions related to logical operations combined with ready-made instructions to extract and deposit bit fields and single cycle multifunction instructions can be used to realize bit stream formatting in a compute inexpensive way.

5 Results and Conclusions

A real-time implementation of MPEG-2 AAC-LC coder on the ADSP 21160 platform, which supports all channel configurations (upto 5.1 channels) has been achieved. This implementation provides broadcast quality [7] compressed audio at 72kbps per channel and above. The utilization factor per frame was 85% in the worst case and 55 – 60 % in the average cases for 5.1 channel encoding (sampling frequency of 48 kHz). This realization supports 32, 44.1 & 48 kHz sampling frequencies. This solution has been developed as a system board (generates Packetized Elementary Stream) to be used for broadcast applications.

The available computational power is optimally used for each channel configuration modes such that the best quality is ensured without using any external memory. It has been found that even modules like bit counting etc. can be efficiently realized on this architecture. The huge on-chip memory aids in reduction of computations via table lookups. A cost effective single chip solution for MPEG-2 AAC-LC coding is possible with ADSP 21160.

References

[1] ADSP-2106x SHARC User's Manual.(Available on-line from www.analog.com)

[2] ADSP-2116x Hammerhead SHARC User's Manual. (Available on-line from www.analog.com)

[3] Marina Bosi, K. Brandenburg, Schuyler Quanckenbush, Louis Fielder, Kenzo Akagiri, Hendrik Fush, Martin Dietz, Jurgen Herre, Grant Davison and Yoshaiki Oikawa. "ISO/IEC MPEG-2 AAC Advanced Audio Coding" Journal of Audio Engg Society Vol. 45, No. 10, Oct. 1997, pp 789-814.

[4] ISO/IEC Standard IS 11172-3 "Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s - Part 3: Audio," 1992.

[5] ISO/IEC Standard IS 13818-3 "Generic Coding of Moving Pictures and Associated Audio Information - Part 3: Audio," 1994.

[6] ISO/IEC 13818-7 "Generic Coding of Moving Pictures and Associated Audio Information - Part 3: (MPEG-2 Advanced Audio Coding, AAC).

[7] ITU-R Recommendation BS.1115, "Low Bitrate Audio Coding," Geneva, Switzerland (1994).