DESIGN OF A LOW POWER PSYCHO-ACOUSTIC MODEL CO-PROCESSOR FOR MPEG-2/4 AAC LC STEREO ENCODER

Tsung-Han Tsai, Shih-Way Huang, Liang-Gee Chen*

Department of Electrical Engineering,
National Central University
han@ee.ncu.edu.tw
DSP/IC Design Lab, Graduate Institute of Electronics Engineering,
Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, ROC
{shihway*, lgchen}@video.ee.ntu.edu.tw

ABSTRACT

A new design of Psycho-Acoustic Model in MPEG-2/4 AAC encoding is proposed. Differing from the conventional PC-based and DSP-based encoders, it was based on hybrid architectures. It was optimized at algorithmic and at architectural levels. At algorithmic level, MDCT-based PAM and fixed-coefficients were used to replace the original FFT-based PAM and spreading functions. At architectural level, the co-processor PAM design is proposed. The proposed design could be used in a real-time MPEG-2/4 AAC LC stereo encoder while maintaining CD quality below 30 MHz.

1. INTRODUCTION

Portable electronic devices such as smart mobile phones, digital cameras, PDAs, and digital audio devices with audio playback have been attractive because of both the prevalence of MP3 audio files and the development of flash memory cards. It is interesting that the so-called MP3's successor, MPEG-2/4 AAC [1], can be equipped in those devices not only for playback but also for recording (encoding). We used ISO MPEG-4 AAC VM at LC stereo, 64 kbps and ran it on a PC with a Pentium III 600 MHz processor. It showed that about 5-times speed-up was needed to achieve real-time encoding. That is, if we would implement it on a portable device, say, Palm-compatible PDA (33 MHz), about 100-times speed-up was needed.

In conventional implementations, real-time AAC encoders are achieved in PC-based or DSP-based [2] platforms with fast algorithms, but they are power-consuming. Y. Takamizawa et al. proposed a good implementation on low power DSP-based AAC encoder [3]. However, due to limited memory capacity, they did not support block-switching mode, which is essential in AAC and gives rise to both better quality and to higher complexity.

The objective is to make a low power, real-time AAC encoder. We will propose a new co-processor design of Psycho-Acoustic Model (abbrev. as PAM), which plays a very important role in AAC, and show how it can lead to a real-time MPEG-2/4 LC stereo encoder with CD quality below 30 MHz.

The paper is organized as follows. In Section 2, we analyzed the AAC algorithms and the PAM algorithms. It is shown the optimization of PAM was necessary and critical. In Section 3, some optimization methods at algorithmic and at architectural levels are proposed to reduce the complexity of PAM. The co-processor architectures are also presented. We finally summarize our major contributions and outline our future work.

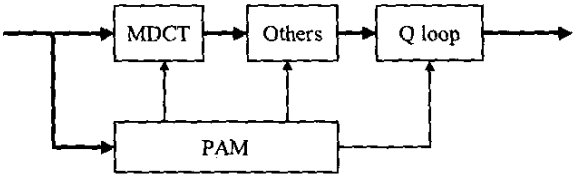

Figure 1. Block diagram of an AAC LC encoder

2. ANALYSES OF AAC ALGORITHMS

While MP3 has been widely used in Internet music distribution and portable devices with audio functions, the so-called successor AAC, reaches on average the same sound quality as MP3 at about 70% of the bitrate [4]. Fig. 1 shows block diagrams of an AAC LC encoder. AAC uses MDCT (Modified Discrete Cosine Transform) to transform input audio samples in time domain into spectrums in frequency domain. The frequency spectrums go on Others, which removes their redundancies by TNS (Temporal Noise Shaping) and joint coding. The spectrums are then quantized and noiselessly coded in the quantization loop (Q loop). PAM, which is usually calculated first, mainly determines the sound quality of a given encoder implementation. PAM outputs the maximum distortion energy that is masked by the signal energy for each coding partition and outputs block types, which determine the block length used in other blocks.

We analyzed the AAC LC stereo encoder in MOPS at the bitrate of 128 kbps and 64 kbps with CD-quality sampling rate of 44.1 kHz and the computational complexity are shown in Table 1.

0-7803-7761-3/03/$17.00 ©2003 IEEE

	128kbps (MOPS)	%	64kbps (MOPS)	%
MDCT	3.0	3.3	3.0	2.0
Q	33.0	35.9	90.1	60.5
PAM	52.5	57.1	52.5	35.2
Others	3.5	3.8	3.5	2.3
total	91.9	100.0	149.0	100.0

Table 1. Computational Complexity Analyses of AAC LC stereo encoder at the bitrate of 128 kbps and 64 kbps

Undoubtedly, PAM and Q loop dominate up to 90% of the computational complexity. From previous works, we know the computational complexity of the Q loop block is in proportion to the number of iterated-loops that is affected by bitrate. About 1/10 number of the original iterated-loops can be reduced by fast and approximate algorithms [5]. Thus, we observed that PAM became the bottleneck in computational complexity of AAC. Since PAM would not be affected by the bitrate, it is in higher proportion at higher bitrate. Hence, PAM had to be sped-up in the low power AAC encoder. In the following section, we will introduce PAM algorithms and its design considerations.

3. ANALYSES OF PAM ALGORITHMS

In the similar way that we analyzed the AAC LC stereo encoder, we went on analyzing PAM. In Fig. 2, we show the PAM algorithms for simplicity according to the 13 steps in [1].

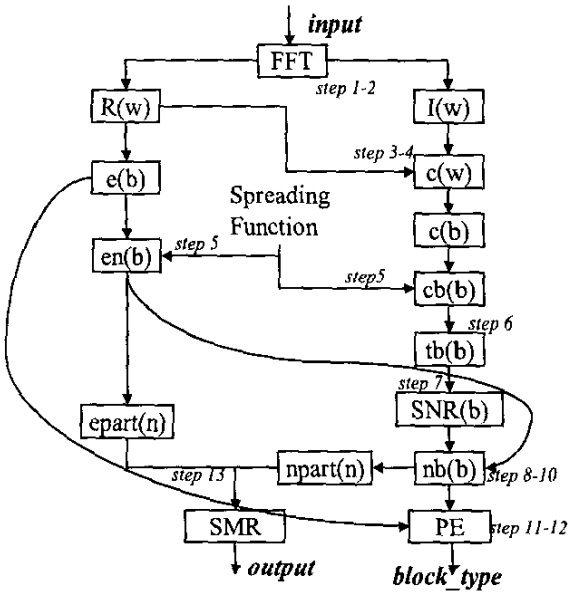

Figure 2. Block diagram of PAM

In steps 1-2, PAM normalizes the time-domain samples as input and transforms them into frequency-domain spectrums of real part R(w) and imaginary part I(w) by FFT. Thereafter, while real-part spectrums lead to the calculation of partitioned energy, imaginary-part spectrums give rise to the calculation of the unpredictability measure c(w) in steps 3-4. In step 5, both partitioned energy and unpredictability are convolved with the spreading function in order to estimate the effects across the partitioned bands. Tonality index is estimated in step 6 to indicate if a signal is tonal-like. SNR (Signal-to-Noise Ratio) is calculated in step 7 and then the masking partitioned energy threshold nb(b) is estimated in steps 8-10 and thus the masking curve is estimated. PE is calculated in steps 11-12 to determine the block type. Finally, SMR (Signal-to-Mask Ratio) is computed in step 13 as output. From Table 2, we observed that step 2, step 4, and step 5 were in high computational complexity. Step 2 and step 4 include some special functions such as arctangent, sine/cosine, square root, etc., which are hard to implement. Step 5 includes spreading functions calculations and convolutions, and was of the highest computational complexity. Thus, we optimized the three steps and present in next section.

step	MOPS	%	step	MOPS	%
1	0.09	0.2	**8**	0.10	0.2
2	6.71	12.8	**9**	0.03	0.1
3	0.71	1.3	**10**	0.10	0.2
4	3.00	5.7	**11**	0.04	0.1
5	40.28	76.8	**12**	0.00	0.0
6	0.10	0.2	**13**	1.14	2.2
7	0.14	0.3	**total**	52.46	100.0

Table 2. Computational Complexity Analyses of PAM

4. DESIGN AND OPTIMIZATION OF PAM

Being well known, it is vital to reduce dynamic power consumption in low power digital design. From [6], the dynamic power consumption is in proportional to factors of effective capacitance, supply voltage, and the clock frequency.

$$P = C_{eff} V_{dd}^{2} f$$

It is now commonly agreed that low power design requires optimizations at all levels of the design hierarchy, i.e. technology, devices, circuits, logic, architecture, and algorithm levels. The reduction of effective capacitance and supply voltage is usually made at the lower levels. Here, we first focus on lowering the clock frequency at the higher levels, i.e. algorithmic and architectural levels. Furthermore, supply voltage may be lowered because of the slower speed and thus power consumption would be dramatically reduced.

4.1 Algorithmic level

- Reduce Calculations of Spreading Function as Fixed-coefficients.

In step 5, calculations of spreading functions are a series of complex functions such as comparisons, square roots, power of tens, squares, and divisions. We found that they were calculated at the square number of partitioned bands. Besides, they were repeatedly estimated every frame. This was the reason that step 5 consumed a lot of MOPS in computational complexity. We also found that the results of spreading functions were only determined by sampling rates and block types. Hence, we reduced the calculations of spreading functions as fixed coefficients, which can be implemented in dedicated ROM with special memory addressing methods.

- MDCT-based PAM.

While there is one main filterbank MDCT outside the PAM, there is another filterbank FFT inside the PAM transforming input samples into spectrums in similar ways. Besides, as has been noted, step 2 and step 4 include some special functions such as arctangent, sine/cosine, square root, etc., which are of high computational complexity and are hard to implement. Here, referring to [3], we replaced FFT with MDCT so that the MDCT outside could be omitted for decreasing computational complexity. Steps 2-4 were thus replaced with MDCT. Step 5 required some modification that only the partitioned energies were convolved with the spreading functions mentioned above because there was no phase-information. Step 6 was also modified where SFM (Spectral Flatness Measure) [7] was used to generate the tonality index from the MDCT coefficients. According to the analyses used above, the two optimization methods could reduce the computational complexity by 78%, which is shown in Table 3.

	MOPS	%
original PAM	52.5	100.00
proposed PAM	11.6	22.1

Table 3. Comparison of original PAM and the proposed PAM in MOPS. After algorithmic optimizations, the computational complexity of PAM was reduced by 78%.

4.2 Architectural level

After using the two optimization methods above, we evaluated that MDCT-based PAM was below 12 MOPS in computational complexity (reduced by 78%), and both Others and Q loop were below 14 MOPS according to the previously mentioned assumption. As for conventional encoders, they could be implemented either by PC-based or by DSP-based platforms. Furthermore, we observed that both Others and Q loop calculated values from current frame, while PAM estimated masking information from next frame. They could be processing at the same time. Therefore, we proposed a new design based on hybrid architectures that comprised a dedicated PAM as a co-processor and a general DSP with low computing power that process Others and Q loop. The dedicated PAM and the general DSP could be processed in parallel or in pipeline with a double-frame-size in-place memory or a FIFO buffer.

5. SIMULATION RESULTS

In order to prove that the modified MDCT-based PAM could work well as the original one, we first simulated our design with several audio files such as castanets, speech, velvet, fatboy, and some from Sound Quality Assessment Materials. Then we compared the sound quality in ODG [8] and NMR with those of the original encoder. From Fig. 3, the positive values of ODG mean that the proposed design is worse; the negative values of ODG mean that the proposed design is better. The largest positive value is below 0.003, which means the difference is very small. In Fig. 4, the positive values of NMR mean that the proposed design is better; the negative values of NMR mean that the proposed design is worse. In most cases, it shows the proposed design is slightly better. There is nearly no degrading between the proposed design and the original encoder.

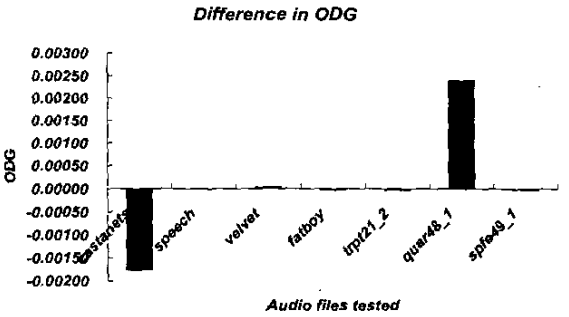

Figure 3. Comparison of sound quality. Difference in ODG of proposed design from that of the original encoder. The positive values of ODG mean the proposed design is worse; the negative values of ODG mean the proposed design is better.

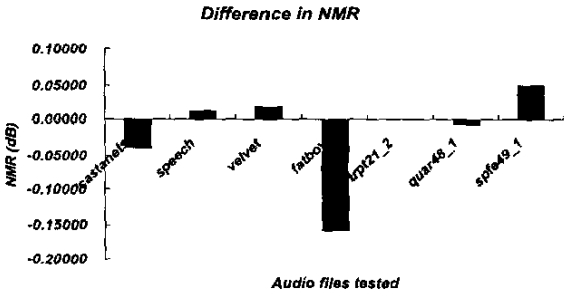

Figure 4. Comparison of sound quality. Difference in NMR of proposed design from that of the original encoder. The positive values of NMR mean the proposed design is better; the negative values of NMR mean the proposed design is worse.

Fig. 5 shows the proposed architectures of AAC. The computational complexities of blocks of optimized algorithms are on the left, and the proposed architectures are on the right side. PAM plays as a co-processor and PAM and Q are processing at the same time (in parallel or in pipeline). Input samples are read into the MDCT-based PAM first. PAM then

calculates and puts the results to the frame memory and then calculates the next frame. The Q+Others get the parameters such as block type and SMR from the PAM and start to calculate the data from frame memory. The working clock rate of the design was estimated below 30MHz. The scheduling diagram is shown in Fig. 6. In Table 4, we compared the DSP-based AAC encoders with the proposed design. Here, we estimated the processing power. From the power formula, the power is in proportional to the clock frequency and thus to MOPS. We calculated them with MDCT-based PAM (12 MOPS), Q (10 MOPS), Others (4 MOPS), assuming the un-optimized PAM (53 MOPS) in the encoder of [2], and then normalized them. Finally, we show the block diagram of the MDCT-based PAM in Fig.7.

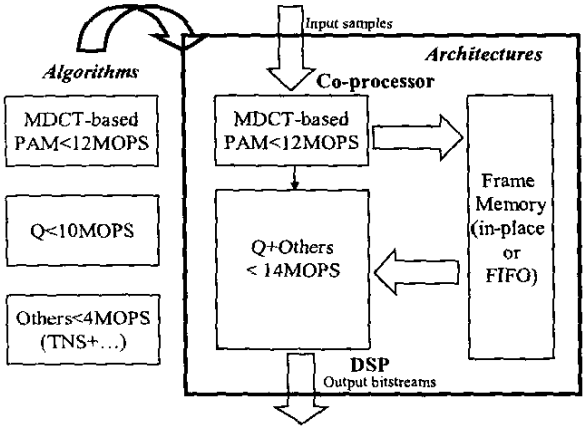

Figure 5. Block diagram of AAC. PAM and Q+Others are in parallel

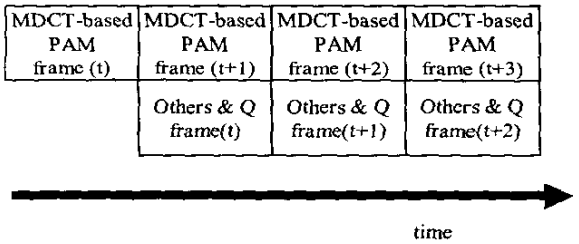

Figure 6. Scheduling of MDCT-based PAM and Others + Q.

	Ref. [2]	Ref. [3]	Proposed
MDCT	optimized	omitted	omitted
Q	optimized	optimized	optimized
PAM	n/a	MDCT-based	MDCT-based
Architecture	DSP-based	DSP-based	Hybrid (ASIC+ DSP)
Processing time	T(MDCT)+T(Q) +T(PAM) +T(Others)	T(Q)+T(PAM) +T(Others)	T(Q) +T(Others)
Processing power	5.0	1.9	1.0

Table 4. Comparison between DSP-based AAC encoders and proposed design

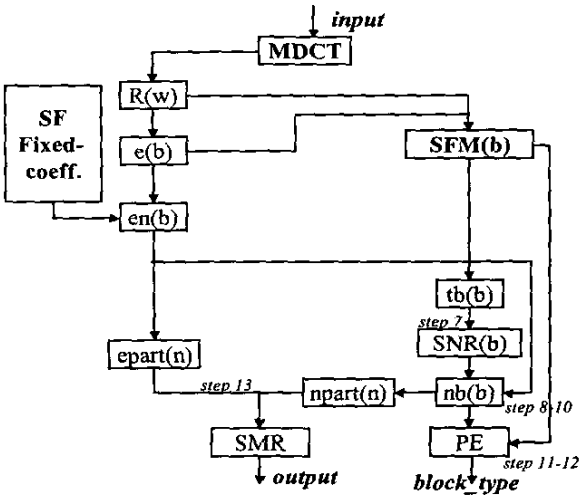

Figure 7. Block diagram of MDCT-based PAM. Note that steps 2-6 are replaced with MDCT and SFM, and fixed coefficients are substituted for calculations of spreading functions.

6. SUMMARIES AND FUTURE WORK

In the paper, we proposed a co-processor MDCT-based PAM so that low power AAC can be achieved by lowering the clock rate to below 30MHz while maintaining CD quality. We optimized the PAM by substituting fixed coefficients for calculations of spreading functions and by replacing the FFT-based PAM with MDCT-based PAM at algorithmic level. We also sped up the AAC by the hybrid architectures of MDCT-based PAM and Others and Q loop in parallel. In the future, we will focus on the implementation of the proposed PAM and extend it to a low power AAC codec.

7. REFERENCES

[1] ISO/IEC 14496-3, "Information technology – Coding of audio-visual objects – Part 3: Audio, " Dec. 1999

[2] Dong-Yan Huang, Xuesong Gong, Daqing Zhou, Toshio Miki, Sanae Hotani, "Implementation of the MPEG-4 advanced audio coding encoder on ADSP-21060 SHARC", *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems*, Vol 3, page(s): 544 -547

[3] Yuichiro Takamizawa, Tsuyoshi Okumura, Toshiyuki Nomura, Masao Ikekawa, and Ichiro Kuroda, "20mW MPEG-2/4 AAC LC Stereo Encoder on a 16-bit DSP", *Workshop and Exhibition on MPEG-4*, San Jose, California, June 25-27 2002,

[4] Karlheinz Brandenburg, "MP3 and AAC Explained", *AES 17th International Conference on High Quality Audio Coding*, Italy, Sep 2-5, 1999

[5] Chi-Min Liu, Chin-Ching Chen, Wen-Chieh Lee, Szu-Wei Lee, "A Fast Bit Allocation Method for MPEG Layer III", *IEEE International Conference on Consumer Electronics, 1999*, pages 22-23.

[6] Jan M.Rabary and M.Pedram, "Low Power Design Methodologies", *Kluwer Academic Publicher Press*, 1996

[7] James. D. Johnston, "Transform Coding of Audio Signals Using Perceptual Noise Criteria", *IEEE Journal on Selected Area on Communications*, Vol. 6, No 2, Feb. 1988.

[8] ITU-T Recommendation BS. 1387: "Method for objective measurements of perceived audio quality," July 2001.