

# LOW POWER FPGA SOLUTION FOR DAB AUDIO DECODER

R.Amutha<sup>1</sup>, PR.Bhuvaneshwari<sup>2</sup>

<sup>1</sup>PG Scholar, ECE Department, Saveetha Engineering College, Tamilnadu, India

<sup>2</sup>Assistant Professor, ECE Department, Saveetha Engineering College, Tamilnadu, India

## Abstract

A new approach to design a DAB audio decoder is introduced to improve the quality of audio at the receiver end. Integrating an MPEG-1 Layer II (MP2) decoder and Advanced Audio Coding Low Complexity (AAC LC) decoder provides basic audio decoding for DAB in FPGA. The audio frames data are generated from DAB channel decoder are stored in RAM. The bit stream Demultiplexer parses the quantized spectrum data in the audio frame and stores them in to the audio RAM. The inverse quantization block reads the quantized spectra from the audio RAM, performs the inverse quantization computations, and writes back the result to the audio RAM. The synthesis filter reads the inverse quantized spectra from the audio RAM, generates the time domain Pulse Code Modulation (PCM) samples, and writes them back to the audio RAM. The whole projects run by using XILINX 12.1 ISE. Finally analyze the power consumption and area occupied by the designed architecture.

**Keywords:** DAB-Digital Audio Broadcasting, AAC LC-Advanced Audio Coding Low Complexity, IQ-Inverse Quantizer, IMDCT-Inverse Modified Discrete Cosine Transform, FPGA-Field Programmable Gate Array

-----\*\*\*-----

## 1. INTRODUCTION

Radio broadcasting is one of the most widespread electronic mass media comprising of hundreds of providers, thousands of HF (High Frequency) transmitters and billions of radio receivers worldwide. The new digital radio system, Digital Audio Broadcasting (DAB) has the capability to replace the existing AM and FM audio broadcast services in many parts. Digital audio data need to be compressed for transmission. Digital audio compression allows the efficient storage and transmission of audio data. The basic task of a perceptual audio coding system is to compress the digital audio data in such a way that the compression is efficient as possible, i.e. the compressed file is as small as possible and the reconstructed (decoded) audio sounds exactly (or as close as possible) to the original audio before compression. Other requirements for Audio compression techniques include low complexity. Perceptual encoding is a lossy compression technique, i.e. the decoded file is not a bit exact replica of the original digital audio data.

### 1.1 DAB Audio Decoder

#### AAC LC Decoder

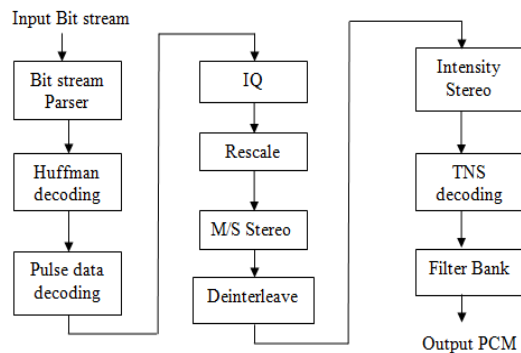
With the advantages on compression ratio and audio quality, MPEG AAC has been used for a wide range of applications from Internet audio to digital audio broadcasting and multichannel surround sound. Even in handheld products such as mobile phones or music players, they support the decoding of AAC files. However, there are several complex algorithms used in the AAC decoding flow which make it difficult to implement as an efficient design. Moreover, power

consumption is also an important issue, particularly for portable devices. There are three different profile defined in AAC. They are the main profile, the low-complexity (LC) profile, and the scalable sampling-rate profile. It allows tradeoffs in audio quality and encoding/decoding complexity for different applications. AAC provides an audio signal that has CD quality at 96-128 kbps/stereo and bit rate 30% lower than that of MP3. Among the three profiles, the LC profile can provides nearly as high audio quality as the main profile but with significant savings in memory and processing requirements.

In AAC audio coding, there are two kinds of audio transport formats. One is Audio Data Interchange Format (ADIF), and the other is Audio Data Transport Stream (ADTS). For the ADIF format, it puts all data controlling the decoder (like sampling frequency, mode, etc.) into a single header preceding the actual audio stream. It is useful for file exchange but does not allow for break-in or start of decoding at any point in time. For the ADTS format, it packs AAC data into frames with headers and allows decoding to begin in the middle of an audio bit stream. AAC is the most advanced MPEG standard for digital audio compression. The design flow of the AAC LC profile is shown in Fig.1. There are several tools used in AAC decoding, which include bit stream parser, Huffman decoding, pulse data decoding, Inverse Quantizer (IQ), rescale, Middle/Side (M/S) and Intensity Stereo, Temporal Noise Shaping (TNS), and Filter bank.

The first block is the bit stream parser, which extracts the audio frame signals and the decoding information that are used in the following decoding tools. In Huffman decoding, there

are 12 Huffman codebooks. Eleven codebooks are used for spectrum coding and one for scale-factor coding. There are two stages in spectral Huffman decoding. Stage 1 is the Huffman decoding which unpacks the Huffman code index. Stage 2 is the regrouping of 2 or 4-tuples of signed or unsigned code words into quantized spectral coefficients. Regrouping processing is performed using the algorithmic (division) approach. Among the 11 spectral Huffman codebooks, book 11 is a special case. It permits the encoding of quantized spectral coefficients even when their largest absolute value (LAV) is larger than 15. If the decoded value equals 16, an escape flag is used to signal the presence of a so called escape sequence.



**Fig-1:** Design Flow of AAC LC profile

The escape sequence consists of an escape prefix of N-bits 1, an escape separator of 1-bit 0, and an escape word of N+4 bits.

The actual decoded values of the escape sequence are  $2^{(N+4)}$  and escape words. Because the input maximum value of inverse quantization is 8191, the maximum length of the escape sequence is 21 bits. The next block is pulse data decoding which uses a pulse amplitude method to represent values larger than 15. When this block is utilized, one or several quantized coefficients are replaced by coefficients with smaller amplitudes in the encoder. In reconstructing the quantized spectral coefficients, these replacements are compensated by adding or subtracting amplitude from the previously decoded coefficients. The quantized values are inversely quantized by the IQ tool and then scaled by the rescale tool. The inverse quantized samples are rescaled by a scale-factor gain. The M/S and intensity stereo tools recover and add the redundancies removed from the encoder to the frequency domain.

In M/S decoding, the left and right channel signals are dematrixed by either the identity matrix or the inverse M/S matrix. When the identity matrix is used, no computation is necessary. Otherwise, the channel pair must be dematrixed to get stereo channel signals. In intensity stereo, it identifies regions in a channel pair which are similar except for their position. Therefore, only left channel data and some additional information to reconstruct the right channel are coded. To recover the right channel signals, it must multiply the Scale

factor. The TNS tool controls the temporal shape of the quantization noise in the frequency domain. It can be applied to the entire spectrum or only a part of the spectrum. In particular, it can use several filters on distinct coefficient regions. Therefore, an all-pole filter is required to select groups of coefficients in the spectral domain. In the filters bank block, the frequency domain signals are transformed into time domain and produce the output audio signals. In contrast to the hybrid filter bank of MP3, AAC uses a plain Modified Discrete Cosine Transform (MDCT). Together with the increased window length (2048 instead of 1152 lines per transformation), the AAC filterbank outperforms the filter banks of previous coding methods. In addition, the long and short windows can be switched dynamically to adapt to the signal properties.

In order to smooth the transition between long and short windows, two transition windows, namely, long-start and long-stop windows, are used. AAC decoder filterbank consists of an Inverse MDCT (IMDCT) and windowing and overlap add functions. Since the windowing function has a significant effect on the filterbank frequency response, the filterbank has been designed to allow the switch in window shape to adapt to input-signal conditions. AAC decoder needs the process that extracts frequency components from a time domain signal. Most of the digital audio decoders use IMDCT to convert the frequency domain signal to the time domain signal. The MP2 decoder is then designed to make it a similar structure to the AAC LC. It consists of the specific blocks of the AAC LC Decoder and other blocks can be shared with the MP2 decoder. For each block, it always reads data from the audio RAM and writes the results back to a different area of the same RAM, just as the AAC LC decoder.

## 2. ARCHITECTURAL DESCRIPTION

MP2 and AAC LC decoder are both based on the psychoacoustic model for audio compression using a time-to-frequency transformation. So even if two audio decoders have different decoding algorithms, their decoding flow charts are similar. As shown in Fig. 2.

It can be seen that both audio decoders have the following blocks bit stream demultiplexer, inverse quantization block, and synthesis filter. These blocks take up most of the computation process for the two audio decoders and lead to low power consumption. The audio frame data are generated from the DAB channel decoder and stored in RAM. A bit payload module conducts the task of reading the desired bits from RAM and transferring them to the MP2 decoder. The bit stream demultiplexer parses the quantized spectrum data in the audio frame and stores them into the audio RAM via the audio RAM interface. The inverse quantization block reads the quantized spectra from the audio RAM, performs the inverse quantization computations, and writes back the result to the audio RAM.



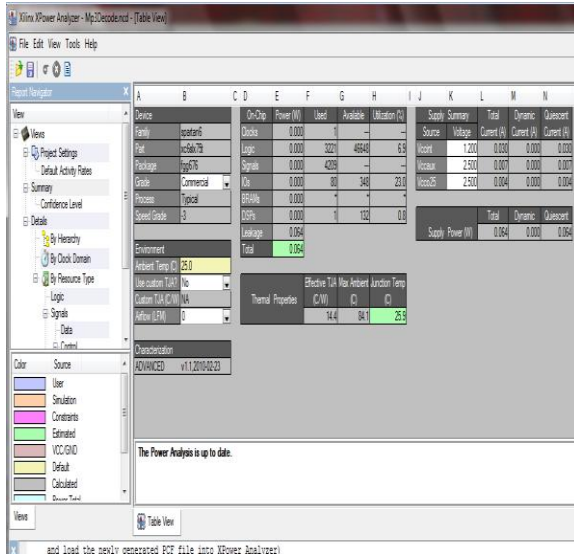


Fig- 4: Power analysis report

#### 4. CONCLUSIONS AND FUTURE WORK

Thus the real time MP2 and AAC decoding system supporting LC profile for DAB audio decoder was designed. The output of this project shall be processed by many blocks before hear the audio services. To verify the designed system by using simulation. The result of simulation showed in both Xilinx ISE 12.1 ISim and MATLAB R2010a software provide the same decoded output. Then the power consumption and area occupied by the architecture have been analyzed and at the same time maintaining the high audio quality. Future work of this project is to implementing this designed architecture using FPGA SPARTAN6. And also measure the audio quality of AAC LC decoder by using PSNR.

#### REFERENCES

- [1]. Guoyu Wang, Hongsheng Zhang, Mingying Lu, Chao Zhang, Tao Jiang and Guangyu Guo, 'Low-cost Low-power ASIC solution for both DAB+ and DAB audio decoding' IEEE Transactions on VLSI systems, pp. 1-9, 2013.
- [2]. Chao-Tang Yu, Kai-Sheng Yang and Yu-Pin Chang, 'DAB Channel Decoder Implementation Using FPGA and Its Testing Platform Buildup', IEEE Transaction on VLSI system, pp. 424-429, 2005.
- [3]. Herre J., 'Temporal noise shaping, quantization and coding methods in perceptual audio coding' AES 17th International Conference High Quality Audio Coding, pp. 312-315, 1999.
- [4]. Jae-Sik Lee, Jong-Hoon Jeong and Tae-Gyu Chang, 'An Efficient method of Huffman decoding for MPEG-2 AAC and its performance analysis' IEEE Transactions on Speech and Audio Processing, vol. 13, No. 6, pp. 1206-1209, 2005.
- [5]. Lu M. Zhang H. Wang G. Guo G. and Jiang T., 'Optimization and hardware implementation of the AAC inverse quantization algorithm for portable DAB+ receivers'

International Journal Digital Content Technology, vol. 6, No. 6, pp. 125-132, 2012.

[6]. Shen-Chuan Tai, Chuen-Ching Wang and Chih-Ying Lin, 'FFT and IMDCT circuit sharing in DAB receiver' IEEE Transactions on Broadcasting, vol. 49, No. 2, pp. 124-131, 2003.

[7]. Tsung-Han Tsai, Chun-Nan Liu, Hsueh-Yi Lin, Hsing-Chuang Liu and Chia-Ying Wu, 'A 1.4 MHz 0.21 mW MPEG-2/4 AAC single chip decoder' IEEE Custom Integrated Circuits Conference (CICC), pp. 653-656, 2009.

[8]. Tsung-Han Tsai, Chun-Nan Liu, 'Low power system design for MPEG-2/4 AAC audio decoder using pure ASIC approach' IEEE Transactions on circuits and systems, vol. 56, pp. 144-155, 2009.

[9]. Yan M. Hu G. and Wang J. (2009) 'Design and implementation of MPEG-4 AAC decoder on ARM embedded system for CMMB receiver' International Conference Manage. Service Sci., pp. 1-3.

[10]. L. Joohyun, K. Sungdo, and K. Jinkyu, "A 159.2 mW SoC implementation of T-DMB receiver including stacked memories," in Proc. IEEE Custom Integr. Circuits Conf., Sep. 2008, pp. 679-682.

[11]. H. Zhang, M. Lu, and G. Wang, "An ASIC implementation of MPEG audio decoders," in Proc. 7th Int. Conf. ASIC, Oct. 2007, pp. 754-757.

[12]. P. Liu, L. Liu, N. Deng, X. Fu, J. Liu, Q. Liu, G. Zhang, and B. He, "VLSI implementation for portable application oriented MPEG-4 audio codec," in Proc. IEEE Int. Symp. Circuits Syst., May 2007, pp. 777-780.

[13]. M. Lu, H. Zhang, G. Wang, T. Jiang, and G. Guo, "Optimization and hardware implementation of the AAC inverse quantization algorithm for portable DAB+ receivers," Int. J. Digit. Content Technol. Appl., vol. 6, no. 6, pp. 125-132, Apr. 2012.

[14]. M. C. Hans and V. Bhaskaran, "A compliant MPEG-1 layer II audio decoder with 16-B arithmetic operations," IEEE Signal Process. Lett., vol. 4, no. 5, pp. 121-122, May 1997.

[15]. D. Schulz, "Improving audio Codecs by noise substitution," J. Audio Eng. Soc., vol. 44, no. 7, pp. 593-598, Jul. 1996.

[16]. T.-H. Tsai and C.-N. Liu, "Low-power system design for MPEG-2/4 AAC audio decoder using pure ASIC approach," IEEE Trans Circuits Syst I, Reg. Papers, vol. 56, no. 1, pp. 144-155, Jan. 2009.

[17]. K. Seong-Do and O. Seung-Hyeub, "A CMOS single chip receiver with RF front-end and baseband processor for band-III T-DMB/DAB applications," in Proc. 11th Int. Conf. Adv. Commun. Technol., pp. 1238-1241, 2009.

[18]. M. Yan, G. Hu, and J. Wang, "Design and implementation of MPEG-4 AAC decoder on ARM embedded system for CMMB receiver," in Proc. Int. Conf. Manage. Service Sci., 2009, pp. 1-3.

[19]. C. W. Ryu, D. H. Lee, H. J. Chi, K. S. Kwan, T. H. im, and J. S. Park, "Design of digital audio DSP core," in Proc. 1st Int. Forum Strategic Technol., Oct. 2006, pp. 59-62.

[20]. D. Zhou, P. Liu, J. Kong, Y. Zhang, B. He, and N. Deng, "An SOC based HW/SW co-design architecture for multi-standard audio decoding," in Proc. IEEE Asian Solid-State Circuits Conf., Nov. 2007, pp. 200–203

## BIOGRAPHIES



**R. Amutha** received the B.E. Degree in Electronics and Communication Engineering in the year 2009 and pursuing M.E. degree in VLSI Design in Saveetha Engineering College. Her area of interest include communication and Low power VLSI Design.



**Mrs. PR. Bhuvaneswari** received the B.E. degree in Electronics and Communication Engineering College and M.E. degree in VLSI Design form Anna University, Chennai in 2005 and 2010 respectively. Currently she is an Assistant Professor in Department of Electronics and Communication Engineering, Saveetha Engineering College, Chennai. Her interests includes Communication Engineering, Electronic Circuit and Computer Networks.