Implementation of MPEG-4 Audio Decoder

Pratiksha R. Wankhade Nitin N. Mandaogade

Abstract— **In this paper we have implemented MPEG-4 AAC decoder. The AAC audio coding is an international standard first be created in MPEG-2 AAC. and is the base MPEG-4 genera audio coding.AAC support up to 48 audio channel and sample rate supported range from 8khz to 96khz. The AAC known as high efficiency advanced audio coding (HE-AAC) which is a part of MPEG-4 audio is also adopted into digital radio standard. Todays widely used the advanced audio coding like MPEG-4 audio decoder.MPEG-4 audio surpasses conventional audio codec scheme in terms of sound quality and compression rate.MPEG-4 AAC decoder is used the audio standard and getting more popular for commercial used.**

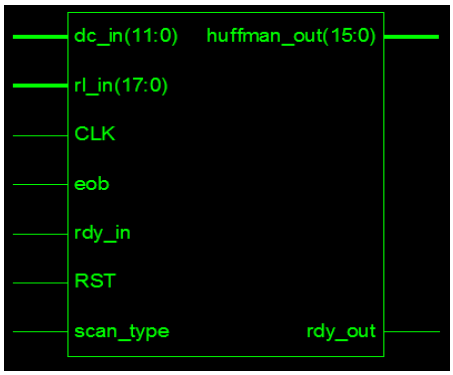
Index Terms- **Huffman Decoder for MPEG-4, IMDCT module, MPEG-4 AAC Decoder**

I. INTRODUCTION

MPEG-4 AAC is the broadly popular as an audio standard and gradually is better accepted for commercial use. MPEG-4 Advanced Audio Coding includes numerous pioneering methodologies so as to attain good audio Fidelity even at lower bitrates.MPEG-4 AAC Decoder is the widely used audio standard and getting more popular for commercial use. The modified discrete cosine transforms (MDCT)/inverse MDCT (IMDCT) is an important block used to realize the MPEG 4 AAC Decoder. This transform is approved by a number of international standards. Though, the computation of the MDCT and IMDCT in the MPEG audio coder could engage a much complex hardware computation. Therefore, to increase the processing bit rate for the Audio decoder there is a need to calculate the Inverse and forward MDCT so that it is obviously appropriate for firmware and hardware implementations.

II. IMPLEMENTATION FOR MPEG 4 AAC DECODER

A. *Huffman Decoder for MPEG 4 AAC*



Huffman Decoder

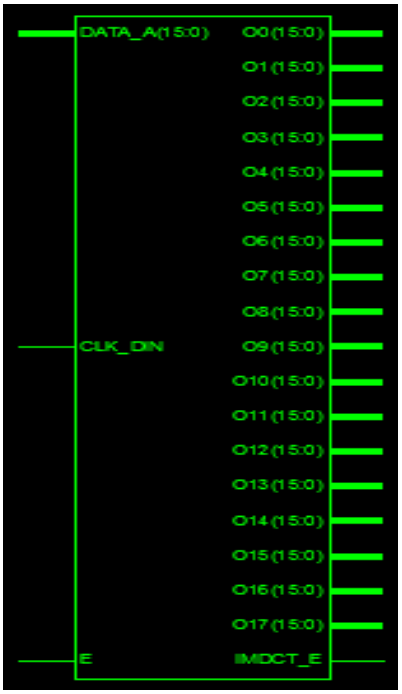
B. *IO Description*

Huffman Decoder			
Port	IO	Size	Description
CLK	I	1	Clock signal for Huffman decoder entity
RST	I	1	Asynchronous Reset signal for the module
eob	I	1	End of module
rdy_in	I	1	Ready signal
scan_type	I	1	Type of scan input to the module
dc_in	I	12	12 bit DC value input
rl_in	I	18	18 bit run length input
huffman_out	O	16	Output of the Huffman decoder
rdy_out	O	1	Output ready signal

C. *Description*

Input of a Mpeg 4 AAC decoder module is compressed bit stream. Prior to interpret or recover the original data we need to decode the incoming bit stream. For that purpose we are using Huffman decoder. Huffman decoder will decode the input compressed data and will provide input to the IMDCT block.

III. MEMORY MODULE (At the input of the IMDCT module)



Memory A

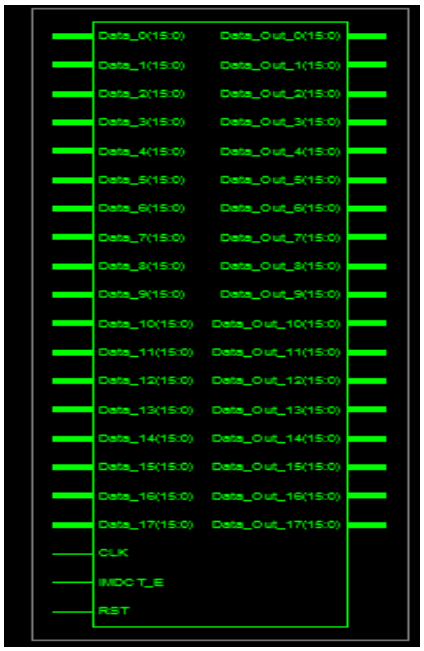
A. IO Description

Memory Module A			
Port	IO	Size	Description
CLK_DIN	I	1	Clock signal for Memory
DATA_A	I	16	16 bit Data input
E	I	1	Enable signal
O(0-17)	O	16	18,16 bit output port applied to the IMDCT module
IMDCT_E	O	1	IMDCT enable signal

B. Description

Output of the Huffman decoder can be fed to the IMDCT module directly because IMDCT will require simultaneous 18, 16 bit inputs. So we have used a memory which will arrange the incoming data from huffman decoder and then will provide simultaneous inputs to the IMDCT module.

IV. IMDCT MODULE



IMDCT Module

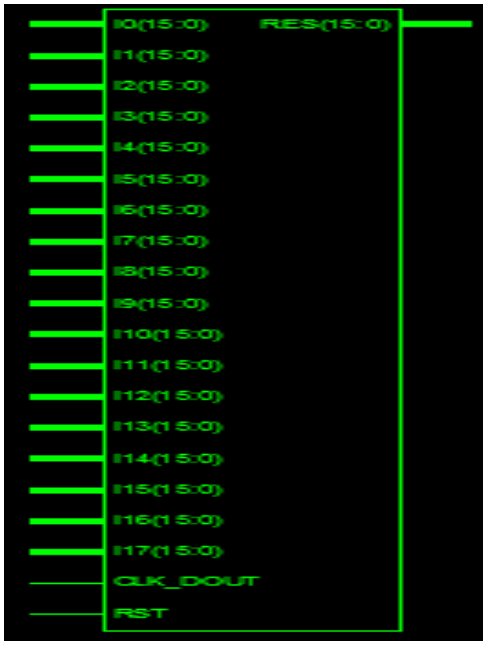
A. IO Description

IMDCT Module			
Port	**IO**	**Size**	**Description**
CLK	I	1	Clock signal for IMDCT module
RST	I	1	Asynchronous reset signal for IMDCT module
IMDCT_E	I	1	Enable signal for IMDCT module
Data_(0-17)	I	16	18, 16-bit input for IMDCT module
Data_Out_(0-17)	O	16	18, 16-bit output for IMDCT module

B. Description

IMDCT module will take the inputs from huffman decoder. This module implements 18 point IMDCT module. It takes 18, 16-bit inputs and after processing it will provide 18, 16-bit output.

V. MEMORY MODULE B (AT THE OUTPUT OF THE IMDCT MODULE)



Memory Module B

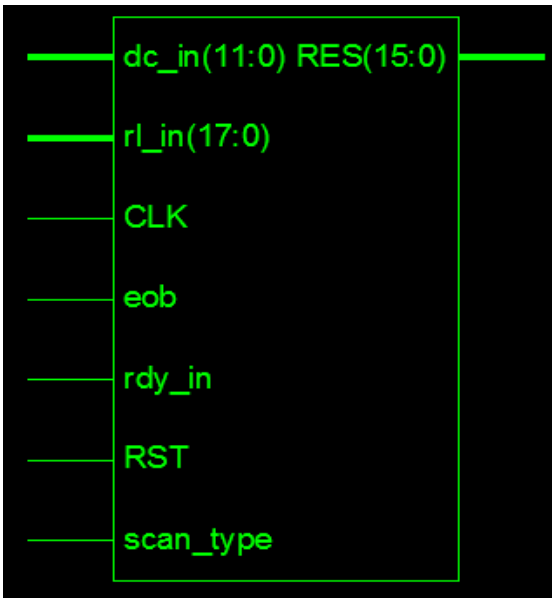
A. IO Description

Memory Module B			
Port	**IO**	**Size**	**Description**
CLK_DOUT	I	1	Clock signal for the module
RST	I	1	Asynchronous Reset signal
I(0-17)	I	16	18, 16-bit input for the IMDCT module
RES	O	16	16 bit output of the module

B. Description

This module will take input from 18-point IMDCT module and will provide 16-bit output at RES port.

VI. MPEG 4 AAC DECODER

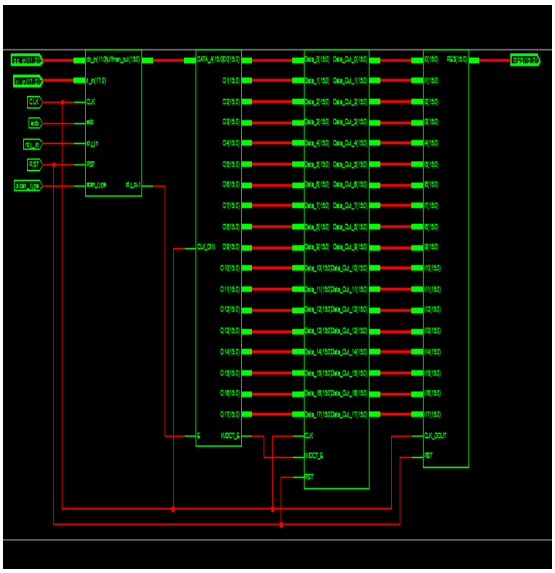


MPEG 4 AAC Decoder Module (Huffman + IMDCT module; Top level entity)

A. Description

After integrating all the blocks we will get the top level module as shown above and the integration will appear as shown below. Inputs are dc_in, rl_in, scan_type,eob, rdy_in, CLK, RST and RES is the 16-bit output.

VII. INTEGRATION OF ALL THE MODULE



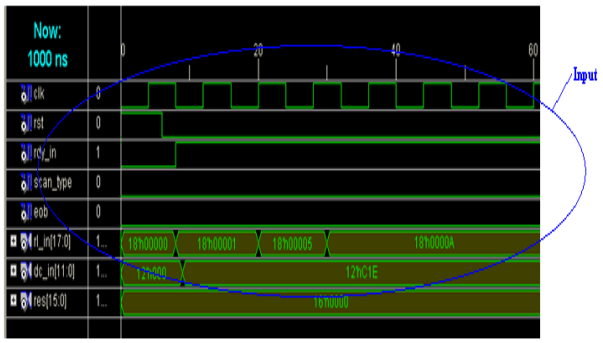
A. *IO Description*

MPEG 4 AAC Decoder			
Port	**IO**	**Size**	**Description**
CLK	I	1	Clock signal for the module
RST	I	1	Asynchronous reset for the module
rdy_in	I	1	Input ready signal
Eob	I	1	End of block signal
scan_type	I	1	Scan type
dc_in	I	12	12 bit Input DC value
rl_in	I	18	18 bit Run length input
RES	O	16	16 bit output

VIII. SIMULATION

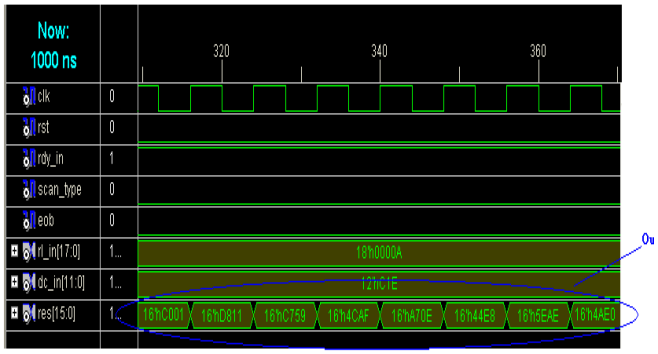
A. *Input to the Decoder*

Above simulation waveform shows the inputs which are provided to the decoder module, these includes CLK, RST, rdy_in, scan_type, eob, rl_in and dc_in.



B. *Output of the Decoder*

Above waveform shows the RES output of the decoder after applying inputs. We will get 16-bit serial output at the RES port.



CONCLUSION

This paper has proposed MPEG-4 audio decoder, which can realize good audio fidelity Even at low bitrates. The modified descrete consine transform(MDCT) and (IMDCT) tnverse transform used to realize the MPEG-4 AAC decoder. Therefore to increase the processing butrate for the audio decoder and also the need to calculated inverse and foemard(MDCT), that is appropriate for firmware and hardware implementation

REFERENCES

[1] ISO/IEC 14496-3,"Information Technology – Very Low Bitrate Audio-Visual Coding Part 3: Audio,'' July 1999.
[2] ISO/IEC JTC1/SC29/WG11 No.1650 "IS 13818-7 (MPEG-2 Advanced Audio Coding, AAC)", April 1997
[3] R.Amutha, P.R.Bhuvaneswari ,"Low power fpga solution for dab audio decoder" , IJRET: International Journal of Research in Engineering and Technology eISSN: 2319-1163 | pISSN: 2321-7308
[4] Matthew A. Watson and Peter Buettner , " Design and implementation of aac decoders" Dolby Laboratories, Inc. San Francisco, CA 94103
[5] Ravindar Sharma, Ghanshyam Kumar Singh, Ram Mohan Mehra " Mp3 Decoder Design & Implementation Using VHDL"
[6] Information Technology - Generic coding of moving picture and associated audio, Part 3: Advanced Audio Coding, ISO/IEC 11172-3
[7] Rassol Raissi "The Theory Behind Mp3" December 2002
[8] Matthew A. Watson and Peter Buettner , "Design and implementation of aac decoders" Dolby Laboratories, Inc. San Francisco, CA 94103
[9] Praveen Sripada ," MP3 DECODER in Theory and Practice"
[10] Fuhui Du, Gaoming Du, Yukun Song, Duoli Zhang, Minglun Gao, " An Implementation of Filterbank for MPEG-2 AAC on FPGA"