ISSN 2229-5518

Fast and Efficient Computation of MDCT/IMDCT Algorithms for MP3 Applications

Sarvesh Babu N S, Asst Prof Y N Mamatha, M K Subramanian

Abstract— The modified discrete cosine transforms (MDCT) and its inverse (IMDCT) has become important for audio codec. MDCT/IMDCT algorithms are widely applied to realize the analysis/synthesis filter banks of the time-domain aliasing cancellation scheme for sub band coding in various audio standards. The inverse modified discrete cosine transform (IMDCT) is the most computational intensive operations in the MPEG audio decoding standard. Efficient algorithms for realizing the IMDCT become very important in decoding process. This paper presents a novel and efficient algorithm for computation of Inverse modified Discrete Cosine Transform (IMDCT) with block size N=12 and 36 points is proposed based on type IV DCT and type II DCT, which improves the efficiency of computation. Firstly IMDCT with block size N is converted into type IV DCT, which is further transformed into DCT II with block size N/4. Comparison of the computational complexity with some known algorithms shows that the proposed new approach reduces the number of arithmetic operations significantly such as number Multiplications and Additions.

Index Terms— Modified Discrete Cosine Transform (MDCT), Inverse Modified Discrete Cosine Transform (IMDCT), type IV Discrete Cosine Transform (DCT-IV), type II Discrete Cosine Transform (DCT-II).

---

1 Introduction

Recent developments of multimedia and communication techniques are rapidly growing. Multimedia systems, which include the video, audio, dynamic picture, image and speech, provide a good environment for people's media exchange and storage. Digital compression of audio signals is widely used in many applications such as transmission speech in the mobile phone, storing music in the memory of MP3 audio music players etc. Typical audio compression schemes are MPEG-1 audio layer III (MP3), MPEG-2 Advanced Audio Coding (AAC), MPEG-4, ATRAC1/ATRAC2 and many others. Most of these audio compression schemes adopted the Modified Discrete Cosine Transform (MDCT) and its inverse Transform (IMDCT). The MDCT and MDST are employed in the sub-band/transform coding schemes as the analysis/synthesis filter bank based on time domain aliasing cancellation (TDAC). The MDCT is the basic processing component for high quality audio compression in typical audio standards. The forward and inverse MDCT are two of the most computationally intensive operations in the well-known audio coding schemes/standards.

In the past, many fast algorithms have been reported in the literature for computing the MDCT and IMDCT. These algorithms can generally be categorized into two kinds: direct method and indirect method. The term of indirect method means that the MDCT or IMDCT is first converted into other unitary transforms such as discrete Fourier transform (DFT) or discrete cosine transform (DCT), and the latter transforms are then calculated by a fast algorithm. These algorithms generally lead to the parallel-in and parallel-out architecture [7, 8, 9, 13]. This is the most widely used technique for the efficient implementation of both MDCT and IMDCT. For example, by decomposing the MDCT kernel and using the symmetry property of cosine function [5,7] presented an efficient MDCT algorithm that is composed of pre-processing (data shifts, differential calculation, and complex pre-multiplication), an N/2-point FFT, followed by complex post-multiplications. In [7], [9], [11] proposed an efficient approach for implementing the MDCT and IMDCT based on the N/4-point DCT/DST and the corresponding N/4–point IDCT/IDST. By using a matrix representation [5] presented a systematic method for realizing the MDCT and IMDCT. A fast algorithm based on the DCT for computing the MDCT and IMDCT is presented [13]. However, a common drawback of the FFT-based method is the need for complex arithmetic and storage of complex values. The disadvantage of the DCT-based method is generally the introduction of recursive structure, which is not suitable for parallel implementation [12]. The direct method for efficiently calculating the MDCT and IMDCT is mainly based on the use of a regressive formula. In [4], [6] proposed a regressive algorithm, which can be implemented by parallel VLSI filters. This algorithm was further improved in [1], [2], [20]. These regressive algorithms in general emphasize on the merits of serial-in and serial-out structures [5], [12].

- *Sarvesh Babu N S, in 4th Semester M Tech, Digital Communication Engineering, in R. V. College of Engineering, Bangalore, India. E-mail: sarveshbabu88@gmail.com*
- *Asst Prof Y N Mamatha, Telecommunication engineering Dept, R. V. College of Engineering, Bangalore, India. E-mail:* mamatharaj_76@rediffmail.com
- *Mr. M K Subramanian, Technical advisor, Davinci Nanotech Pvt. Ltd., Bangalore, India. E-mail: mks@davincinanotech.com*

ISSN 2229-5518

Therefore, efficient algorithms for the MDCT and IMDCT are of prime importance in audio coding and decoding process. Based on subband analysis/synthesis approaches [1]–[4], [7], [12], [17] the modified discrete cosine transforms (MDCT or forward MDCT) and its inverse MDCT or backward MDCT) have extensively been adopted in various audio codec standards [1], [2], [11], [17], [18]. The latest standards adopt a larger transform size, e.g., N= 12 and 36 in MPEG-1 Audio Layer 3 (MP3) [1, 13, 18].

This paper is organized as follows. In Section 2, we introduce the definitions and some properties of the MDCT and IMDCT. Section 3 describes the fast algorithm for the efficient computation of MDCT/IMDCT. The results and analysis of the computational complexity and comparison of the proposed algorithm with some existing ones are given in Section 4. Section 5 concludes this paper.

2 Definition of MDCT /IMDCT and Properties

N-point MDCT and N/2-point IMDCT computations are respectively defined as

$$X(k) = \sum_{n=0}^{N-1} x(n) \cos\left\{\frac{\left(2n+1+\frac{N}{2}\right)(2k+1)\pi}{2N}\right\} \qquad (1)$$

$$for\ k = 0, 1, . . . \ , N/_2 - 1.$$

$$\bar{x}(n) = \sum_{k=0}^{\frac{N}{2}-1} X(k) \cos\left\{\frac{\left(2n+1+\frac{N}{2}\right)(2k+1)\pi}{2N}\right\} \qquad (2)$$

$$for\ n = 0, 1, \ . . . \ , N - 1.$$

The input sequence x(n) in equation (1) is assumed to be windowed by the sine windowing function before its transformation. MDCT is equivalent to the modulated lapped transform. $\bar{x}(n)$ in equation (2) represents the time domain aliased data sequence recovered by the Inverse or backward MDCT (IMDCT) block transform which does not correspond to original data sequence x(n).

The 36-IMDCT that produces 36 output values from 18 input values. These samples are multiplied with a 36-point window before it can be used by the next step in the decoding process. The window to use is based on the block type that can be found in the side information. Producing 36 samples from 18 frequency lines means that only 18 of the samples are unique. So the IMDCT use a 50% overlaps. The 36 values from the windowing are divided into two groups, a low group and a high group, containing 18 values each. Overlapping is performed by adding values from the lower group with corresponding values from the higher group from the previous frame.

In case of long blocks the IMDCT generates an output values for every 18 input values which produces 36 output values. In case of short blocks three transforms are performed which produces 12 output values each. The vectors are windowed and overlapped with each other. Concatenation 6 zeros on both ends of the resulting vector gives a vector a vector of length 36, which is processed like the output of a long transform

3 Fast Algorithm for MDCT/IMDCT Computation

3.1 Realization of MDCT/IMDCT

In this section, we describe fast computation of MDCT by decomposing N-point MDCT into two MDCT of length N/4. Then IMDCT algorithm can be obtained by simply reversing last N/4 point input and then performing N/4-point IDCT-II. From this section window length is assumed to be divisible by 4, i.e. $\grave{N} = N/4$.

A new universal fast rotation-based MDCT computational structure based on fast rotation-based DCT-IV computational structure is described in [15]:

The data sequence y_n in forward MDCT computation is given by,

$$y_{N/_4+1} = x_n - x_{N/_2-1-n}\ , \qquad (4a)$$

$$y_{N/_4-1-n} = -x_{N/_2+n} - x_{N-1-n}, \quad for\ n = 0, 1, . . . \ , N/_4 - 1 \qquad (4b)$$

The Jacobi rotations for $N/_4$ block length is given by

$$a_n = y_n \cos\frac{\pi}{2N}(2n+1) + y_{\frac{N}{2}-1-n} \sin\frac{\pi}{2N}(2n+1), \qquad (5a)$$

$$b_n = -y_n \sin\frac{\pi}{2N}(2n+1) + y_{\frac{N}{2}-1-n} \cos\frac{\pi}{2N}(2n+1), \qquad (5b)$$

$$for\ n = 0, 1, . . . \ , N/_4 - 1$$

The fast rotation-based IMDCT computational structure for k odd/even and specifically for $\grave{N} = N/4$ is defined as,

$$\bar{y}_{2k} = \sum_{n=0}^{\frac{\bar{N}}{2}-1} \left[\left(a_n + a_{\frac{N}{4}-1-n}\right) \cos\varphi_{k,n} + \left(b_n + b_{\frac{N}{4}-1-n}\right) \sin\varphi_{k,n}\right]$$

$$\bar{y}_{\frac{N}{2}-1-2k} = \sum_{n=0}^{\frac{\bar{N}}{2}-1} \left[(-1)^{n+1}\left(-1\left\{a_n + a_{\frac{N}{4}-1-n}\right\}\right) \sin\varphi_{k,n} + \left(b_n + b_{\frac{N}{4}-1-n}\right) \cos\varphi_{k,n}\right]$$

IJSER

$$\bar{y}_{\frac{N}{2}-2k} = \sum_{n=0}^{\frac{\acute{N}}{2}-1}\Big[(-1)^{n}\Big(-1\Big\{b_n + b_{\frac{N}{4}-1-n}\Big\}\Big)\cos\varphi_{k,n} + \Big(a_n + a_{\frac{N}{4}-1-n}\Big)\sin\varphi_{k,n}\Big]$$

$$\bar{y}_{2k-1} = \sum_{n=0}^{\frac{\acute{N}}{2}-1}\Big[\Big(-1\Big\{b_n + b_{\frac{N}{4}-1-n}\Big\}\Big)\sin\varphi_{k,n} + \Big(a_n + a_{\frac{N}{4}-1-n}\Big)\cos\varphi_{k,n}\Big] \quad (6)$$

Where $\varphi_{k,n} = \frac{\pi}{2\acute{N}}(2n+1)k, \quad for\ n = 0, 1, \ldots, {}^{N}/_{4} - 1.$

$for\ k = 1, \ldots, {}^{N}/_{4} - 1.$

For k=0, the coefficients $\bar{y}_0$ and $\bar{y}_{\frac{N}{2}-1}$ are given by

$$\bar{y}_0 = a_{\frac{(\acute{N}-1)}{2}} + \sum_{n=0}^{\frac{\acute{N}}{2}-1}\Big(a_n + a_{\frac{N}{4}-1-n}\Big) \quad (7a)$$

$$\bar{y}_{\frac{N}{2}-1} = (-1)^{\frac{\acute{N}-1}{2}+1} b_{\frac{\acute{N}-1}{2}} + \sum_{n=0}^{\frac{\acute{N}}{2}-1}(-1)^{n+1}\Big[b_n + b_{\frac{N}{4}-1-n}\Big] \quad (7b)$$

3.2 Implementation of IMDCT basic block

Since for MP3 N = 12 or 36, input sequence X_k to the IMDCT block will be $\{X_0, X_1, \ldots, X_{N/2-1}\}$ as shown in Fig.1.

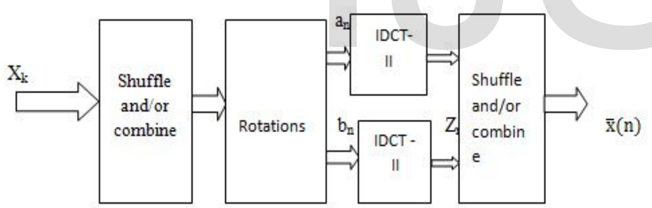

Fig. 1. Architecture of N/2-point IMDCT using two N/4-point DCT-II

The output of leftmost shuffel and/or combine and rotation block is calculated by equation (5). By using equation (6) and equation (7) compute the values of $\bar{y}_{N/2}$. The time domain aliased data sequence $\bar{x}(n)$ in inverse MDCT(IMDCT) can be recovered by using fllowing equations:

$$\bar{x}_n = y_{\frac{N}{4}+n} \quad (8a)$$

$$\bar{x}_{\frac{N}{2}-1-n} = -y_{\frac{N}{4}+n} \quad (8b)$$

$$\bar{x}_{\frac{N}{2}+n} = -y_{\frac{N}{4}-1-n} \quad (8c)$$

$$\bar{x}_{N-1-n} = -y_{\frac{N}{4}-1-n} \quad (8d)$$

$$for\ n = 0, 1, \ldots, {}^{N}/_{4} - 1$$

3.3 DCT-IV based IMDCT implementation for N=36

Based on complete equations (5)-(7) the DCT-IV based computational structure IMDCT is calculated. Signal graph for computation of IMDCT for N=36 is shown in Fig. 2.

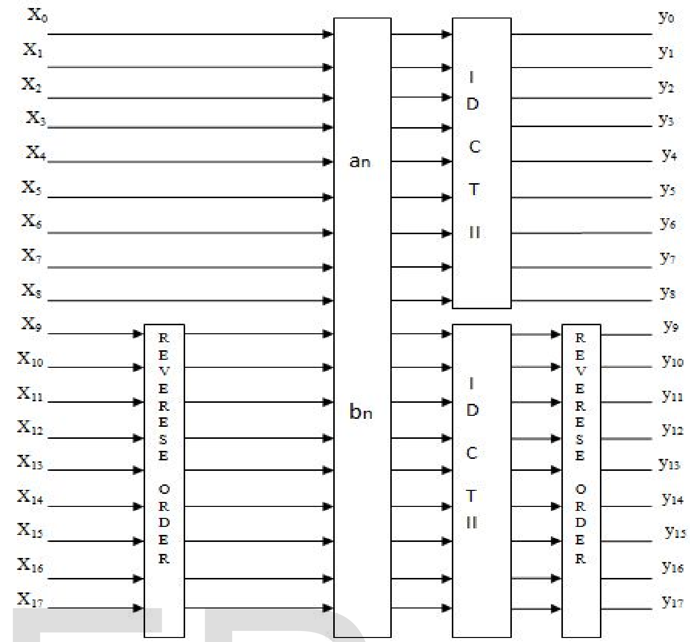

Fig. 2. Signal graph for efficient computation of IMDCT for N=36.

Once the values of $\bar{y}_n$ is calculated, by using equation (8) 36-point IMDCT values can be obtained for $\bar{y}_n$. Similar procedure is been used for N=12. The resulting computational complexity for block size N=12 is 11 multiplications and 18 additions and for block size N=36 is 43 multiplications and 102 additions as shown in Table I. Plot of frequency spectrum for standard test input to MP3 decoder according to MPEG compliance specification with frequency range from 20Hz to 22 KHz is shown in Fig. 3. For standard input error that obtained in IMDCT values by fast computation method is 9.926167E-23 to 0.0000 when compared with direct method. The RMS level of the difference signal and Maximum absolute value of the difference signal is calculated as [20]:

$$RMS\ of\ difference\ signal = \sqrt{\frac{1}{N}\left(\sum_{i=1}^{N}(t_i - r_i)^2\right)} \quad (9)$$

$$absolute\ difference\ signal = |t_i - r_i| \quad (10)$$

Where, t_i is the i^{th} output sample from the decoder under test, r_i is the i^{th} sample from reference output and N is the number of samples.

IJSER

4 Results

TABLE I. COMPARISON OF EFFICIENT IMPLEMENTATIONS OF THE IMDCT BLOCK TRANSFORMS IN MP3 IN TERMS OF THE ARITHMETIC COMPLEXITY.

IMDCT algorithm	N=12 Points	N=36 points
Britanak's	11M + 21A + 2S	43M + 129A + 6S
DIT Mixed radix	15M + 29A + 2S	67M + 147A + 6S
Improved DIF mixed radix	21M + 43A + 4S	110M + 213A + 24S
Lee's algorithm	11M + 23A + 2S	43M + 115A + 6S
Truong's	11M + 23A + 2S	43M + 115A + 6S
Radix-3	20M + 42A	84M + 174A
Our Algorithm	11M + 18A + 2S	43M + 102A + 4S

M-real Multiplications, A-real Additions, S-Shift.

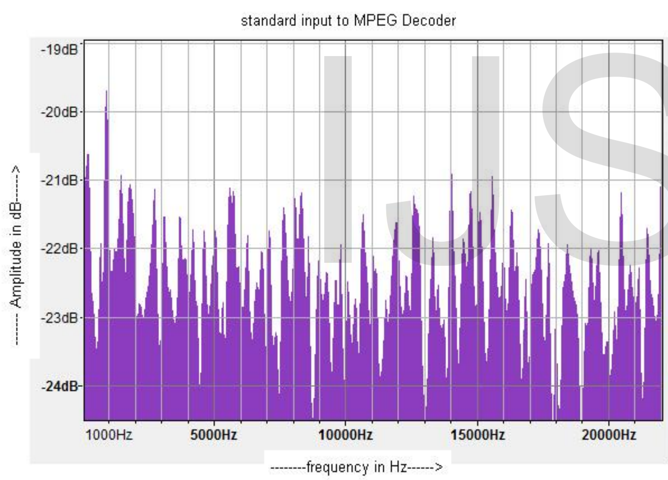

Fig. 3a. Frequency specturm plot for standard test input to the MP3 Decoder according to MPEG Compliance specifications.

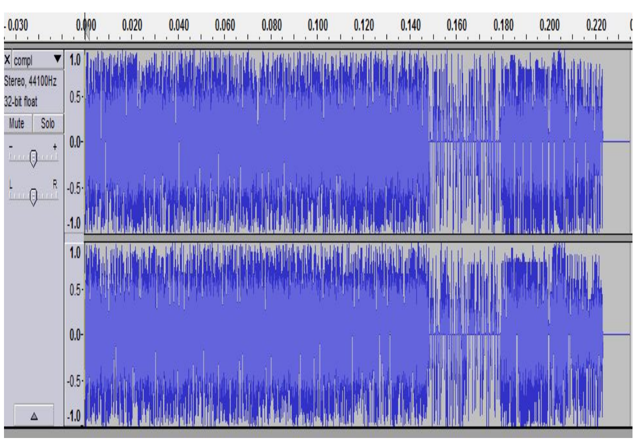

Fig. 3b. Encoded MP3 file for standard test input to the MP3 Decoder frequency ranging from 10Hz to 22KHz according to MPEG Compliance specifications[20].

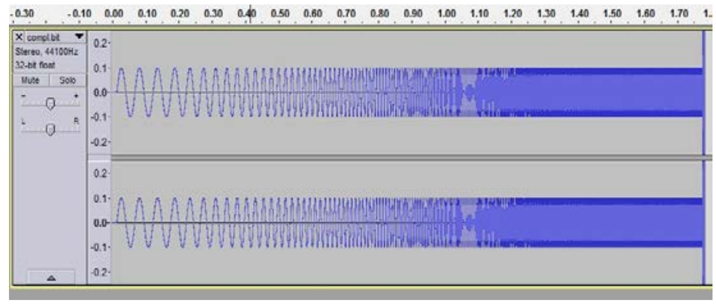

Fig. 3a. MP3 decoder output for standard test input using by direct computation method.

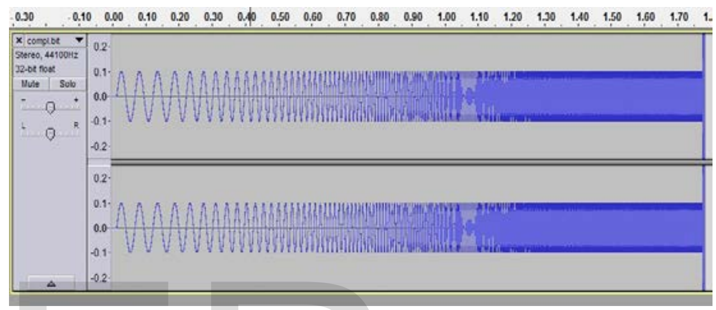

Fig. 3b. MP3 decoder output for standard test input using by using Fast Computation method.

IJSER

Fig. 3a shows the frequency spectrum of standard test input to the MP3 Decoder with frequency range from 10Hz to 22KHz according to MPEG Compliance specifications[20]. Encoded MP3 file for standard test input of Fig. 3a is shown in Fig. 3b. Output of the decoder for standard test input is shown in Fig. 4. By comparing Fig. 4a and Fig. 4b it shows that output of the MP3 decoder by using fast algorithm approach is same as that of direct method. RMS and absolute difference signal is zero is calculated by equation (9) and (10) respectively, for standard input sine sweep signal to the MP3 decoder which satisfies MPEG audio decoder compliance criteria [20].

5 Conclusion

The New computational Structures identical for an efficient implementation of both forward modified discrete cosine transforms (MDCT) and its inverse (IMDCT) has been described. They are based on fast rotation-based MDCT computational structure. For the short and long audio block it is shown that our Algorithm implementation for MP3 can be modified and achieves less arithmetic complexity as shown in Table 1. The time required for execution of our algorithm is reduced to 35% when compared to direct method and it satisfies MPEG MP3 decoder compliance specifications

.

REFERENCES

[1] Sheau-Fang Lei, Shin-Chi Lai, Po-Yin Cheng, and Ching-Hsing Luo, "Low Complexity and Fast Computation for Recursive MDCT and IMDCT Algorithms," *IEEE Transactions on Circuits and Systems—II: Express Briefs*, vol. 57, no. 7, July 2010.

[2] Shin-Chi Lai, Sheau-Fang Lei, and Ching-Hsing Luo, "Common Architecture Design of Novel Recursive MDCT and IMDCT Algorithms for Application to AAC, AAC in DRM, and MP3 Codecs," *IEEE Transactions on Circuits and Systems—II: Express Briefs*, vol. 56, no. 10, October 2009.

[3] Jiasong Wu, Huazhong Shu, Lotfi Senhadji and Limin Luo, "Mixed-Radix Algorithm for the Computation of Forward and Inverse MDCTs," *IEEE Transactions on Circuits and Systems: Regular Papers*, vol. 56, no. 4, April 2009.

[4] Vladimir Nikolajevic and Gerhard Fettweis, "Computation of Forward and Inverse MDCT Using Clenshaw's Recurrence Formula," *IEEE Transactions on Signal Processing*, vol. 51, no. 5, may 2003.

[5] Mu-Huo Cheng and Yu-Hsin Hsu, "Fast IMDCT and MDCT Algorithms—A Matrix Approach," *IEEE Transactions on Signal Processing,* vol. 51, no. 1, January 2003.

[6] Che-Hong Chen, Bin-Da Liu and Jar-Ferr Yang, "Recursive Architectures for Realizing Modified Discrete Cosine Transform and Its Inverse," *IEEE Transactions on Circuits and Systems—II: Analog and Digital Signal Processing,* vol. 50, no. 1, January 2003.

[7] Vladimir Britanak and K. R. Rao, "An Efficient Implementation of the Forward and Inverse MDCT in MPEG Audio Coding," *IEEE Signal Processing Letters*, vol. 8, no. 2, February 2001.

[8] Szu-Wei Lee, "Improved Algorithm for Efficient Computation of the Forward and Backward MDCT in MPEG Audio Coder," *IEEE Transactions on Circuits and Systems—II: Analog and Digital Signal Processing,* vol. 48, no. 10, October 2001.

[9] Vladimir Britanak, K. R. Rao, "A New Fast Algorithm for the Unified Forward and Inverse MDCT and MDST computation," *Signal Processing 82 (2002)* 433 – 459, *Elsevier Science*, 8 June 2002.

[10] Huazhong Shu, Xudong Bao, Christine Toumoulin and Limin Luo, "Radix-3 Algorithm for the Fast Computation of Forward and Inverse MDCT," *IEEE*, *IEEE Signal Processing Letters*, vol. 14, no. 2, February 2007.

[11] Thuong Le-Tien, Vu Cao-Tuan, Chien Hoang-Dinh, "FPGA-BASED ARCHITECTURE OF MP3 DECODING CORE FOR MULTIMEDIA SYSTEMS," *Hochiminh City University of Technology*, 2007.

[12] Vladimir Britanak, "A Survey of Efficient MDCT Implementations in MP3 Audio Coding Standard: Retrospective and state-of-the-art," *Elsevier Science, Signal Processing 91 (2011)* 624–672, 22 September 2011.

[13] T.K. Truong, P.D. Chen, T.C. Cheng "Fast Algorithm for Computing the Forward and Inverse MDCT in MPEG Audio coding," *Elsevier Science, Signal Processing 86 (2006),* 1055–1060, 31 August 2005.

[14] Z.G. Gui, Y.Geb, D.Y.Zhang, J.S.Wu, "Generalized Fast Mixed-radix Algorithm for the computation of Forward and Inverse MDCTs," *Elsevier Science Signal Processing 92 (2012)*, 363–373, 9 August 2011.

[15] Vladimir Britanak, "New fast computational structures for an efficient implementation of the forward/backward MDCT in MP3 audio coding standard", Elsevier Science, Signal Processing 90 (2010) 536–547, 30 July 2009.

[16] Vladimir Britanak, "The Fast DCT-IV/DST-IV computation via the MDCT, *In Short communication, Elsevier Science, Signal Processing 83 (2003)*, 1803 – 1813, 6 February 2003.

[17] Xingdong Dai, Meghanad D.Wagh, "An MDCT Hardware Accelerator for MP3 Audio," *IEEE Symposium on Application Specific Processors (SASP 2008)* ,1-4244-2334-7/08, 121-125, 2008.

[18] Shin-Chi Lai, Yi-Ping Yeh, Wen-Chieh Tseng, and Sheau-Fang Lei, "Low-Cost and High-Accuracy Design of Fast Recursive MDCT/MDST/IMDCT/IMDST Algorithms and Their Realization," IEEE *Transactions on Circuits And Systems—II: Express Briefs*, Vol. 59, No. 1, January 2012.

[19] Peter Malík, "Computational Models Designed in MATLAB to Improve Parameters and Cost of Modern Chips," *Institute of Informatics, Slovak Academy of Sciences, Slovak Republic, www.intechopen.com*

[20] http://www.underbit.com/resources/mpeg/audio/compliance.

[21] Hi-Seok Kimand and Hackyoon Kim, "FPGA Implementation of MDCT/IMDCT Core in Audio Coding Standards," *Journal of Advanced Information Technology and Convergence*, VOL. 2, NO. 1, April 2012.

IJSER