Available online at www.sciencedirect.com

ScienceDirect

AASRI Procedia 5 (2013) 177 – 182

AASRI Procedia

www.elsevier.com/locate/procedia

2013 AASRI Conference on Parallel and Distributed Computing and Systems

A New Recursive Decomposition Algorithm to Calculate IMDCT

Hui Li[a],*, Yiwen Wang[a], Ping Li[a], Yunyi Li[a]

[a] State key Laboratory of Electronic Thin Films and Integrated Devices, University of Electronic Science& Technology of China, Chengdu (610054), China

Abstract

A new decomposition method to calculate IMDCT is proposed in this paper. The decomposition algorithm can convert a pair of long transforms into 2 pairs of short transforms with half size of the long transforms. In addition, the decomposition algorithm can be utilized recursively to attain shorter transforms and reduce computational cycles. According to experimental results obtained previously, the number of computational cycles of 512-point IMDCT is estimated to be 9641, which is less than those in previous reports.

Keywords: IMDCT; DCT-IV/DST-IV; TDAC

1. Introduction

MDCT and IMDCT have been utilized to implement TDAC [1] in many standards. Because calculating MDCT/IMDCT requires much time, it is necessary to find efficient methods and simple structures for the task.

To reduce area cost, the regressive methods and architectures were researched [2-6]. Allowing for VLSI implementation, the methods proposed in [2,3] are same efficient. But the architecture in [2] has a latch more. The method in [4] reduces about 7 eighths of computational time compared with ones in [2,3] by virtues of

* Corresponding author. Tel.: +86-28-83206813; fax: +86-28-83208839.
E-mail address: huili@uestc.edu.cn

doi:10.1016/j.aasri.2013.10.075

low latency of IDCT-II and IMDCT's symmetry. In contrast with the method in [4], our factorization method have the virtues of less computational cycles and smaller structure [5]. Absorbing virtues of the algorithm in [4] and [5], our recent proposed algorithm [6] possesses smaller accelerator and less computational cycles in contrast to [5].

A new decomposition method to calculate IMDCT is proposed in this paper. The work presented in this paper results from the method in [5]. Its aim is to further develop the algorithm in [5] and increase the computational efficiency for IMDCT computation. The core idea is to recursively decompose a pair of long transforms into 2 pairs of short transforms with half size of the long transforms. The detailed derivation of algorithm is proposed in section II. Conclusion is presented in Section III.

2. Decomposition algorithm of IMDCT

2.1. Obtaining two M/4-point transforms from one M-point IMDCT

The equation of M-point IMDCT is expressed as

$$x(m)=\sum_{k=0}^{M/2-1} X(j)\cos\frac{\left[2m+1+\frac{M}{2}\right]\left[2j+1\right]\pi}{2M}\ ,\qquad \text{for m} = 0,\ldots, \text{M} - 1. \tag{1}$$

M is a multiple of 16 in this paper.

Three sequences are defined as

$$y(m)=\begin{cases} x\left(m-M/4\right), & for \quad m=M/4,\ldots,M-1,\\ -x\left(m+3M/4\right), & for \quad m=0,\ldots,M/4-1,\end{cases} \tag{2}$$

$$y^{'}(m)=y(2m), \qquad \text{for } m{=}M/4\text{-1}, \tag{3}$$

$$y^{''}(m)=y(2m+1), \qquad \text{for } m{=}M/4\text{-1}. \tag{4}$$

We can further derive

$$z_{cc}(m)^{(0)}=\sum_{j=0}^{M/2-1} X(j)C(j)_M\cdot\cos[(2m+1)(2j+1)\pi/M], \tag{5}$$

$$z_{cs}(m)^{(0)}=\sum_{j=0}^{M/2-1} X(j)S(j)_M\cdot\sin[(2m+1)(2j+1)\pi/M]\ , \tag{6}$$

where

$$\begin{cases} z_{cc}(m)^{(0)}=[y^{'}(m)+y^{''}(m)]/2, \quad m=0,\ldots,M/2-1,\\ z_{cs}(m)^{(0)}=[y^{'}(m)-y^{''}(m)]/2, \quad m=0,\ldots,M/2-1,\\ C(j)_M=\cos\{[2j+1]/[2M]\}\pi, \quad j=0,\ldots,M/2-1,\\ S(j)_M=\sin\{[2j+1]/[2M]\}\pi, \quad j=0,\ldots,M/2-1.\end{cases} \tag{7}$$

From (3),(4), (7), we have

$$y(2m)=z_{cc}(m)^{(0)}+z_{cs}(m)^{(0)}, \tag{8}$$

$$y(2m{+}1)=z_{cc}(m)^{(0)}-z_{cs}(m)^{(0)}, \tag{9}$$

where m ranges from 0 to *M*/4 -1. Thus, the two M/4-point transforms ($z_{cc}(m)^{(0)}$ and $z_{cs}(m)^{(0)}$) are used to calculate the *M*-point IMDCT.

2.2. Decomposing two M/4-point transforms into four M/8-point transforms

The two $M/4$-point transforms presented in part A will be decomposed into four $M/8$-point transforms in this part. Then, the four $M/8$-point transforms will be recombined to get four new $M/8$-point transforms. And the relation between the transforms and $y(m)$ is given. The derivation is as follows. Let

$$Z_{cc}(j)^{(0)} = X(j)C(j)_M\,, \tag{10}$$

$$Z_{cs}(j)^{(0)} = X(j)S(j)_M\,. \tag{11}$$

Dealing with equations (5),(6),(7),(10) and (11) elaborately, we derive

$$z_{cc}(m)^{(1)} = \sum_{j=0}^{M/2-1} Z_{cc}(j)^{(0)}C(j)_{M/2}\cdot\cos\{[2m+1][2j+1]\pi/[M/2]\}, \tag{12}$$

$$z_{cs}(m)^{(1)} = \sum_{j=0}^{M/2-1} Z_{cc}(j)^{(0)}S(j)_{M/2}\cdot\sin\{[2m+1][2j+1]\pi/[M/2]\}, \tag{13}$$

$$z_{cc}(2m)^{(0)} = z_{cc}(m)^{(1)} + z_{cs}(m)^{(1)},\quad for\quad m = 0,\dots,M/8-1; \tag{14}$$

$$z_{cc}(2m+1)^{(0)} = z_{cc}(m)^{(1)} - z_{cs}(m)^{(1)},\quad for\quad m = 0,\dots,M/8-1; \tag{15}$$

and

$$z_{ss}(m)^{(1)} = \sum_{j=0}^{M/2-1} Z_{cs}(j)^{(0)}C(j)_{M/2}\cdot\sin\{[2m+1][2j+1]\pi/[M/2]\}, \tag{16}$$

$$z_{sc}(m)^{(1)} = -\sum_{j=0}^{M/2-1} Z_{cs}(j)^{(0)}S(j)_{M/2}\cdot\cos\{[2m+1][2j+1]\pi/[M/2]\}, \tag{17}$$

$$z_{cs}(2m)^{(0)} = z_{ss}(m)^{(1)} + z_{sc}(m)^{(1)},\quad for\quad m = 0,\dots,M/8-1; \tag{18}$$

$$z_{cs}(2m+1)^{(0)} = z_{ss}(m)^{(1)} - z_{sc}(m)^{(1)},\quad for\quad m = 0,\dots,M/8-1; \tag{19}$$

Based on the above derivation, computing two M/4-point transforms (equations (9) (10)) is changed into computing four M/8-point transforms(equations (12)(13)(16)(17)) .The relation between y(m) （m =0,1,…,M/2-1）and $z_{cc}(m)^{(1)}, z_{cs}(m)^{(1)}, z_{ss}(m)^{(1)}, z_{sc}(m)^{(1)}$ （m =0,1,…,M/8-1） is given as follows.

From equations (8),(9), (14),(15), (18) and (19) we have

$$y(4m) = [z_{cc}(m)^{(1)} + z_{sc}(m)^{(1)}] + [z_{cs}(m)^{(1)} + z_{ss}(m)^{(1)}],\qquad m = 0,\ \dots,\ M/8-1, \tag{20}$$

$$y(4m+3) = [z_{cc}(m)^{(1)} + z_{sc}(m)^{(1)}] - [z_{cs}(m)^{(1)} + z_{ss}(m)^{(1)}],\qquad m = 0,\ \dots,\ M/8-1, \tag{21}$$

$$y(4m+1) = [z_{cc}(m)^{(1)} - z_{sc}(m)^{(1)}] + [z_{cs}(m)^{(1)} - z_{ss}(m)^{(1)}],\qquad m = 0,\ \dots,\ M/8-1, \tag{22}$$

$$y(4m+2) = [z_{cc}(m)^{(1)} - z_{sc}(m)^{(1)}] - [z_{cs}(m)^{(1)} - z_{ss}(m)^{(1)}],\qquad m = 0,\ \dots,\ M/8-1. \tag{23}$$

Based on equations (12),(13),(16) and (17), equations (20),(21),(22) and (23) can be further changed into

$$y(4m) = z_{c_0}\left(m\right)^{(1)} + z_{s_0}\left(m\right)^{(1)},\qquad m = 0\ ,\ \dots\ ,\ M/8-1, \tag{24}$$

$$y(4m+3) = z_{c_0}\left(m\right)^{(1)} - z_{s_0}\left(m\right)^{(1)},\qquad m = 0\ ,\ \dots\ ,\ M/8-1, \tag{25}$$

$$y(4m+1) = z_{c_1}\left(m\right)^{(1)} + z_{s_1}\left(m\right)^{(1)},\qquad m = 0\ ,\ \dots\ ,\ M/8-1, \tag{26}$$

$$y(4m+2) = z_{c_1}\left(m\right)^{(1)} - z_{s_1}\left(m\right)^{(1)},\qquad m = 0\ ,\ \dots\ ,\ M/8-1. \tag{27}$$

while

$$z_{c_0}\left(m\right)^{(1)} = \sum_{j=0}^{M/2-1} Z_{c_0}\left(j\right)^{(1)}\cdot\cos\{[2m+1][2j+1]\pi/[M/2]\}\ ,\ m = 0\ ,\ \dots\ ,\ M/8-1, \tag{28}$$

$$z_{s_0}(m)^{(1)} = \sum_{j=0}^{M/2-1} Z_{s_0}(j)^{(1)} \cdot \sin\{[2m+1][2j+1]\pi/[M/2]\},\ m=0,\ldots,M/8-1, \tag{29}$$

$$z_{c_1}(m)^{(1)} = \sum_{j=0}^{M/2-1} Z_{c_1}(j)^{(1)} \cdot \cos\{[2m+1][2j+1]\pi/[M/2]\},\ m=0,\ldots,M/8-1, \tag{30}$$

$$z_{s_1}(m)^{(1)} = \sum_{j=0}^{M/2-1} Z_{s_1}(j)^{(1)} \cdot \sin\{[2m+1][2j+1]\pi/[M/2]\},\ m=0,\ldots,M/8-1. \tag{31}$$

and

$$Z_{c_0}(j)^{(1)} = Z_{cc}(j)^{(0)} C(j)_{M/2} - Z_{cs}(j)^{(0)} S(j)_{M/2}, \tag{32}$$

$$Z_{s_0}(j)^{(1)} = Z_{cc}(j)^{(0)} S(j)_{M/2} + Z_{cs}(j)^{(0)} C(j)_{M/2}, \tag{33}$$

$$Z_{c_1}(j)^{(1)} = Z_{cc}(j)^{(0)} C(j)_{M/2} + Z_{cs}(j)^{(0)} S(j)_{M/2}, \tag{34}$$

$$Z_{s_1}(j)^{(1)} = Z_{cc}(j)^{(0)} S(j)_{M/2} - Z_{cs}(j)^{(0)} C(j)_{M/2}. \tag{35}$$

Equations (28)(29)(30)(31) are four new M/8-point transforms. From equations (20)(21)(22)(23) and equations (24)(25)(26)(27) , we can see that $y[m]$ is expressed as sum of 4 items using M/8-point transforms (equations (12)(13)(16)(17)) whereas sum of 2 items using new M/8-point transforms (equations (28)(29)(30)(31)). That is to say, using new M/8-point transforms can decrease computational complexity.

Fig.1. is presented to clarify the relation among $y[m]$ （m =0,1,…,M/2-1）and two M/4-point transforms and four new M/8-point transforms. The box of M/4-point SUM_0 corresponds with equations (8) and the box of M/4-point SUB_0 corresponds with equations (9). The four boxes in right column correspond with equations (24)-(27).

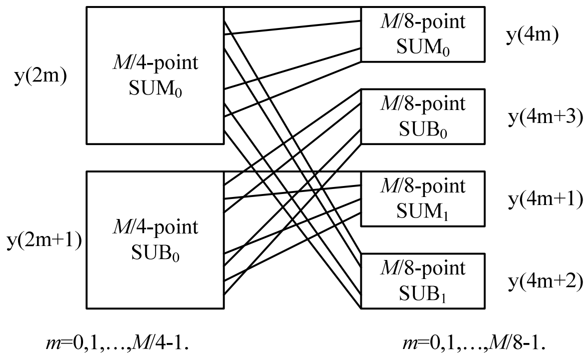

Fig. 1. relation among $y(m)$ and M/4-point transforms and M/8-point transforms

2.3. Conversion to DCT-IV/DST-IV

To reduce the computational cycles, equations (28) (29) (30) (31) are changed into two pairs of M/8-point DCT-IV/DST-IV as

$$z_{c_0}(m)^{(1)} = \sum_{j=0}^{M/8-1} Z_{c_0}(j)^{(2)} \cdot \cos\{[2m+1][2j+1]\pi/[M/2]\},\ m=0,\ldots,M/8-1 \tag{36}$$

$$z_{s_0}(m)^{(1)} = \sum_{j=0}^{M/8-1} Z_{s_0}(j)^{(2)} \cdot \sin\{[2m+1][2j+1]\pi/[M/2]\},\ m=0,\ldots,M/8-1 \tag{37}$$

$$z_{c_1}(m)^{(1)} = \sum_{j=0}^{M/8-1} Z_{c_1}(j)^{(2)} \cdot \cos\{[2m+1][2j+1]\pi/[M/2]\}, \quad m = 0, \ldots, M/8-1 \tag{38}$$

$$z_{s_1}(m)^{(1)} = \sum_{j=0}^{M/8-1} Z_{s_1}(j)^{(2)} \cdot \sin\{[2m+1][2j+1]\pi/[M/2]\}, \quad m = 0, \ldots, M/8-1 \tag{39}$$

while

$$Z_{c_0}(j)^{(2)} = Z_{c_0}(j)^{(1)} + Z_{c_0}(M/2-1-j)^{(1)} - Z_{c_0}(M/4-1-j)^{(1)} - Z_{c_0}(M/4+j)^{(1)} \tag{40}$$

$$Z_{s_0}(j)^{(2)} = Z_{s_0}(j)^{(1)} - Z_{s_0}(M/2-1-j)^{(1)} + Z_{s_0}(M/4-1-j)^{(1)} - Z_{s_0}(M/4+j)^{(1)} \tag{41}$$

$$Z_{c_1}(j)^{(2)} = Z_{c_1}(j)^{(1)} + Z_{c_1}(M/2-1-j)^{(1)} - Z_{c_1}(M/4-1-j)^{(1)} - Z_{c_1}(M/4+j)^{(1)} \tag{42}$$

$$Z_{s_1}(j)^{(2)} = Z_{s_1}(j)^{(1)} - Z_{s_1}(M/2-1-j)^{(1)} + Z_{s_1}(M/4-1-j)^{(1)} - Z_{s_1}(M/4+j)^{(1)}. \tag{43}$$

By using equations (32)-(35), (10) and (11), $Z_{c_0}(j)^{(2)}$, $Z_{s_0}(j)^{(2)}$, $Z_{c_1}(j)^{(2)}$, $Z_{s_1}(j)^{(2)}$ in equations (40)-(43) can be described as functions of $X(j)$ in similar forms as ones in [5]. Equations (24)-(27), (36)-(39) have similar correspondences in [5] too. Therefore, there are similar computational cycles as [5] using equations (24)-(27), (36)-(39) to compute IMDCT. But the decomposition algorithm from two M/4-point transforms to four M/8-point transforms in part B, section II can be used recursively to get eight M/16-point transforms. After that, Equations (36)-(39) will become four pairs of M/16-point DCT-IV/DST-IV and computational cycles are reduced by 50%. Equations (40)-(43) will be in the form of eight items' sum instead of four items' sum and computational cycles are approximately doubled. Namely, the cycles to calculate 512-point IMDCT are approximately 9641 from the experimental results of [5]. Therefore, the presented decomposition algorithm can reduce computational cycles of IMDCT compared with 12153 in [5] and 9718 in [6].

3. Conclusion

A new decomposition method to calculate IMDCT is proposed in this paper. The decomposition algorithm can convert a pair of long transforms into 2 pairs of short transforms with half size of the long transforms. In addition, the decomposition algorithm can be utilized recursively to attain shorter transforms and reduce computational cycles. Based on the previous experimental results, the number of computational cycles of 512-point IMDCT is estimated to be 9641, which is less than 12153 in [5] and 9718 in [6]. The detailed number of computational cycles will be evaluated in future experiments.

Acknowledgements

The research work reported in this paper is supported by "the Fundamental Research Funds for the Central Universities" (Project Code: ZYGX2012J040).

References

[1] J. P. Princen and A. B. Bradley, Analysis/synthesis filter bank designs based on time domain aliasing cancellation, IEEE Trans. Acoust. Speech Signal Process. 1986;34: 1153-1161.
[2] H. C. Chiang and J. C. Liu, Regressive implementation for the forward and inverse MDCT in MPEG audio coding, IEEE Signal Process. Lett. 1996; 4: 116-118.
[3] V. Nikolajevic and G. Fettweis, Computation of forward and inverse MDCT using Clenshaw's recurrence formula, IEEE Trans. Signal Process. 2003; 5: 1439-1444.

[4] C. H. Chen, B. D. Liu and J. F. Yang, Recursive architectures for realizing modified discrete cosine transform and its inverse, IEEE Trans. Circuits Systems-II: Analog Digital Process. 2003; 1: 38-45.
[5] H. Li, P. Li, Y. Wang, et al, A New Decomposition Algorithm of DCT-IV DST-IV for Realizing Fast IMDCT Computation, IEEE Signal Process. Lett. 2009; 9: 735-738.
[6] H. Li, P. Li, Y. Wang, An efficient hardware accelerator architecture for implementing fast IMDCT computation, Signal Process. 2012; 8: 2540-1545.