FPGA BASED EFFICIENT PARALLEL ARCHITECTURE OF LIFTING BASED CDF (2,2) FOR IMAGE COMPRESSION

K. Prashanth Kumar[1], R. C. Joshi[2] and A. K. Saxena[3]
Department of Electronics and Computer Engineering,
IIT Roorkee, Roorkee, India
E-mail- kumarfec@iitr.ernet.in

Abstract

An improved architecture for two-dimensional discrete wavelet transform (2D-DWT) to implement bi-orthogonal Cohen-Daubechies-Feuvear (CDF) (2,2) wavelet with line-based method is proposed for FPGA implementation using lifting scheme. The FPGA based hardware implementation profits especially from the high parallelism in the architecture and the moderate number precision required to preserve the qualitative effects of the mathematical models. The proposed architecture is designed to generate 4 sub bands coefficients concurrently per clock cycle that can perform a 1-level decomposition of a *N x N* image in exactly $N_2 / 4$ working clock cycles, without any line buffers at the column processor, thus reducing the time for line buffering but with an extra row processor and with 100% hardware utilization.

Key words: Cohen-Daubechies-Feuvear (CDF), Discrete Wavelet Transform (DWT), Field Programmable Gate Array (FPGA), Lifting Scheme, Wavelet.

I. INTRODUCTION

The Discrete Wavelet Transform (DWT) has gained widespread acceptance in signal processing and image compression. Because of their inherent multi-resolution nature, DWT has been adopted in JPEG2000 [1] coding stream which has given rise to various DWT algorithms and their VLSI implementations to reduce complexity and enhance performance. In modern hardware design, it's a fact that storage resource is more expensive than computation resource. So the key problem in hardware implementation is to achieve high performance while maintaining low memory requirement. To exploit parallelisms fully is a short cut to achieve high performance, and to reuse data is the way to reduce memory requirement.

Recent research on DWT has focused on a form of lifting which shows excellent performance compared to the conventional convolution method. Factoring discrete wavelet transform into lifting steps can reduce the computational complexity by 50% [2] and has advantages, including integer to integer transform [3], symmetric forward and inverse transforms[4]. Line-based architecture for the direct two-dimensional discrete wavelet transform (2D-DWT) is an efficient alternative tradeoff between the speed and area [5],[6]. Field Programmable Gate Arrays (FPGAs) find applications in many areas of digital signal processing especially when high parallelism is offered by architecture. In addition reconfigurability [7] of FPGAs allows the implementation of more than one custom application on a single FPGA.

In this paper an improved architecture to implement bi-orthogonal Cohen-Daubechies-Feuvear (CDF) (2,2) wavelet on FPGA is proposed employing the parallel and pipelined techniques. The proposed architecture can perform a 1-level decomposition of an *N x N* image in exactly $N^2 / 4$ internal clock cycles, without any line buffers at the column processor, thus reducing the time for line buffering but at the cost of additional row processor.

The rest of the paper is organized as follows: In Section 2, lifting scheme and factoring wavelets into lifting scheme and CDF (2,2) wavelet using lifting scheme are briefly reviewed. The proposed architecture is presented in detail in Section 3, then implementation in Section 4. Finally conclusion in Section 5.

II. FACTORING WAVELETS INTO LIFTING SCHEME

A. Lifting Scheme

In 1994, Sweldens proposed a efficient way of constructing the biorthogonal wavelet bases, which he called the lifting scheme [8]. The basic structure of the lifting scheme [9] is shown in Fig.1. The input signal $S_{j,k}$ is first split into an update function to even and odd samples. The detail (i.e., high-frequency) coefficients $d_{j-l,k}$ of the signal are then generated by subtracting the output of a prediction function *P* of the odd samples from the even samples. The smooth coefficients (the low frequency components) are produced by adding the odd samples to the output of an update function *U* of the details. The computation of either the detail or smooth coefficients is called a *lifting step*.

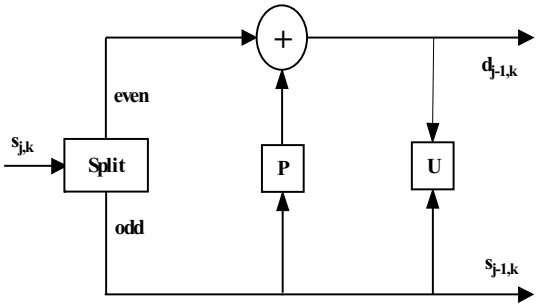

Fig.1. Lifting scheme

B. Factoring Wavelet filters Into Lifting Scheme

Daubenchies and Sweldens [2] showed that every FIR wavelet or filter bank can be factored into a cascade of lifting steps that is, as a finite product of upper and lower triangular matrices and a diagonal normalization matrix. The high pass filter *g(z)* and low pass filter *h(z)* can thus be rewritten as

$$g(z)=\sum_{i=0}^{J-1} g_i z^{-i} \quad (1)$$

$$h(z)=\sum_{i=0}^{J-1} h_i z^{-i} \quad (2)$$

where is the filter length. We can split the high pass and low pass filters into even and odd parts:

$$g(z) = g_e(z^2) + z^{-1} g_0(z^2) \quad (3)$$

$$h(z) = h_e(z^2) + z^{-1} h_0(z^2) \quad (4)$$

The filters can also be expressed as a polyphase matrix as follows:

$$P(z)=\begin{bmatrix} h_e(z) & g_e(z) \\ h_0(z) & g_0(z) \end{bmatrix} \quad (5)$$

Using the Eculidean algorithm which recursively finds the greatest common divisors of the even and odd parts of the original filters, the forward transform polyphase matrix $P_{(z)}$ can be factored into lifting steps as follows:

$$\tilde{P}(z)=\prod_{i=1}^{m}\begin{bmatrix} 1 & 0 \\ -s_i(z^{-1}) & 1 \end{bmatrix}\begin{bmatrix} 1 & -t_i(z^{-1}) \\ 0 & 1 \end{bmatrix}\times\begin{bmatrix} \frac{1}{K} & 0 \\ 0 & K \end{bmatrix}, m \le K \quad (6)$$

where $s_i(z)$ and $t_i(z)$ are Laurent polynomial corresponding to the update and predict steps, respectively, and *K* is a non zero constant. The inverse DWT is described by the following equation:

$$P(z)=\prod_{i=1}^{m}\begin{bmatrix} 1 & s_i(z) \\ 0 & 0 \end{bmatrix}\begin{bmatrix} [1] & 0 \\ t_i(z) & 1 \end{bmatrix}\begin{bmatrix} K & 0 \\ 0 & \frac{1}{K} \end{bmatrix} \quad (7)$$

C. Chohen-Daubechies-Feauveau (CDF) (2,2) Wavelet Using Lifting Scheme:

The analyzing filter pair for the CDF [10] with 2 vanishing moments for both primal lifting and dual wavelet function is (up to a normalization factor of)

$$\tilde{h}(z)=-\frac{1}{8}z^{-2}+\frac{1}{4}z^{-1}+\frac{3}{4}+\frac{1}{4}z-\frac{1}{8}z^{2} \quad (8)$$

$$\tilde{g}(z)=\frac{1}{4}z^{-2}-\frac{1}{2}z^{-1}+\frac{1}{4} \quad (9)$$

Following the above procedure in B we can factor the analysis polyphase matrix of a CDF(2,2) wavelet

$$\tilde{P}(z)=\begin{bmatrix} 1 & 0 \\ 0 & -\frac{1}{2} \end{bmatrix}\begin{bmatrix} 1 & \frac{1}{4}+\frac{1}{4}z \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ -\frac{1}{2}z^{-1}-\frac{1}{2} & 1 \end{bmatrix} \quad (10)$$

The lifting structure for the CDF (2,2) is shown in Fig.2.

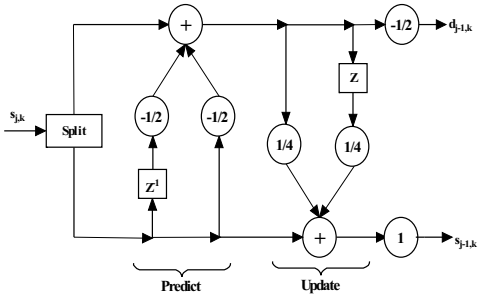

Fig.2. Lifting structure for CDF (2,2) wavelet

III. PROPOSED ARCHITECTURE

An improved parallel architecture for implementing 2D-DWT of CDF (2,2) compared to parallel architecture proposed in [11] is proposed in this paper. The CDF (2,2) is widely used for image compression because of its good compression characteristics. From Eq. (10) if we omit the normalization factors $\frac{1}{K}$ and, it can be implemented by the following algorithm:

Splitting
$$s_i \leftarrow x_{2i}$$
$$d_i \leftarrow x_{2i+1} \quad (11)$$

Dual Lifting
$$d_i \leftarrow d_i - \frac{1}{2}(s_i + s_{i+1}) \quad (12)$$

Primal Lifting
$$s_i \leftarrow s_i + \frac{1}{4}(d_{i-1} + d_i) \quad (13)$$

The proposed architecture as shown in Fig.3, is composed of three row wise1-D DWT modules (R_WT1, R_WT2 and R_WT3) and two column wise 1-D DWT modules (C_WT1 and C_WT2). In normal CDF (2,2) implementation, first all the rows are processed then all the columns are processed. So, for parallel processing column processor not only requires present two processed row elements but also next processed row which is clear from Eq.(12) and Eq.(13); in order to supply future row processed elements for column processor, extra row processor is used which not only avoids extra line buffering at column processor but also decomposes the image in exactly $N^2/4$ cycles. Therefore, nine input samples are required simultaneously in each internal working clock cycle with four sub bands coefficients generated synchronously. R_WT1 module requires even-row-even-column, even-row-odd column, even-row-next even column and R_WT2 module requires odd-row-even-column, odd-row-odd column, odd-row- next even column and R_WT3 requires next even-row-even-column, next even-row-odd column, next even-row-next even column. R_WT1, R_WT2 and R_WT3 row processors operates in parallel and produces low frequency and high frequency components simultaneously, then column processors C_WT1 and C_WT2 takes low frequency components and high frequency components from row processors respectively and generates LL,LH,HL and HH components in the same clock cycle. The architecture of row processors can be designed by directly mapping the lifting factorization of CDF (2,2) wavelet, which can also be extended to that of column processor.

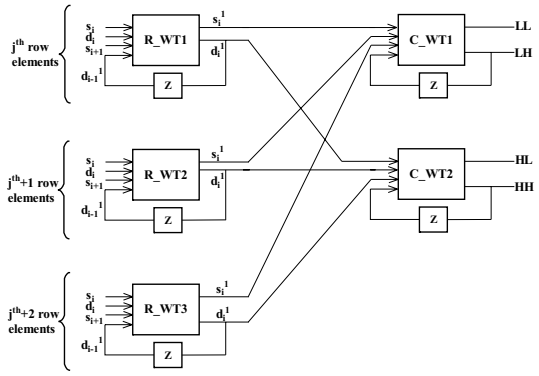

Fig.3. Proposed architecture

IV. IMPLEMENTATION

Initially the proposed architecture is tested using MATLAB software, the image format used here is Portable Grey Map (PGM) format in which pixels are stored in unsigned char type, providing a maximum of 256 gray scale levels or 8-bit data per pixel. Original image 'Barbara.pgm'(512 x512) and the image after level 1 decomposition is shown in Fig.4. Then for proposed architecture code is written in VHDL for FPGA implementation. Code is simulated in Modelsim, synthesized in Xilinx ISE and burned on Spartan-3E [12] FPGA kit. Simulations result is shown in Fig.5.

Fig.4. Barbara image and its image after level 1 decomposition

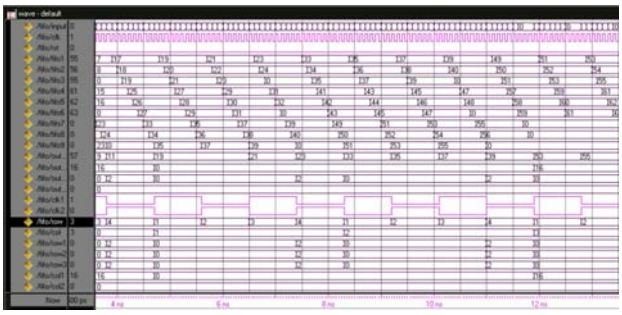

Fig.5. Simulation result for image for 8 x 8 FPGA implementation

V.CONCLUSION

Proposed architecture is tested using MATLAB software and implemented in FPGA using Xilinx ISE. Compared to previous parallel architecture [11], proposed architecture generates four sub band coefficients concurrently in every clock cycle, therefore it can perform level 1 decomposition of a $N \times N$ image in exactly $N^2/4$ working clock cycles, without any need of line buffers at the column processors but with the cost of additional row processor, Hence it has better performance in terms of throughput with reduced memory.

ACKNOWLEDGEMENT

This work is supported by Special Man Power Development in VLSI and related softwares, Phase-II (SMDP-II), Ministry of Information Technology, and Government of India.

REFERENCES

[1] C. Christopoulos, A. Skodras and T. Ebrahimi, 2000, "The JPEG2000 still image coding system: an overview," IEEE Trans. on Consumer Electronics, vol.46, no. 4, pp.1103-1127.

[2] I. Daubechies, W. Sweldens, 1998, "Factoring wavelet transforms into lifting schemes," J. Fourier Anal. Appl., vol.4, pp.247-269.

[3] A .R. Calderbank, I. Daubechies, W. Sweldens, and B.L. Yeo, 1998, "Wavelet transform that map integers to integers," Applied and Computational Harmonic Analysis (ACHA), vol.5, no.3, pp.332-369.

[4] Kishore Andra, Chaitali Chakrabarti and Tinku Acharya, 2002, " VLSI architecture for lifting-based forward and inverse wavelet transform", IEEE Trans. on Signal Processing,vol.50, no.4,pp.966-977.

[5] C. Chrysafis, and A. Ortega, 2000, "Line-based, reduced memory, wavelet image compression," IEEE Trans. on Image Processing, vol.9,no.3, pp.378-389.

[6] P. Wu, and L. chen, 2001, "An efficient architecture for two-dimensional discrete wavelet transform," IEEE Trans. on Circuits and Systems for Tech., vol.11, no.4, pp.536-545.

[7] D. Bhatia, 1997, "Reconfigurable computing," 10th International Conference on VLSI Design, Hyderabad, India, pp. 356-359.

[8] W. Sweldens, 1995, "The lifting scheme: A new philosophy in biorthogonal wavelet constructions," in Proc. SPIE, vol.2569, pp.68-79.

[9] H. Liao, M. M. Mandal, and B. F. Cockburn, 2004, "Efficient architecture for 1-D and 2-D lifting-based wavelet transforms," IEEE Trans. on Signal Processing, vol.52, no.5, pp. 1315-1326.

[10] A. Cohen, I. Daubechies, and J. Feauveau, 1992, "Bi-orthogonal bases of compactly supported wavelets," Comm. Pure Appl. Math., pp.485-560.

[11] Cheng-Yi Xiong, Jin-Wen Tian and Jian Liu, 2005, "Efficient parallel architecture for lifting-based row-dimensional discrete wavelet transform," IEEE Int. Workshop VLSI Design and Video Tech., Suzhou, China, pp.75-78.

[12] Spartan-3E Starter Kit Board User Guide, 2006, UG230 (v1.0) http://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf

K. Prashanth Kumar received his Diploma degree in Electronics and Communications from Govt. Polytechnic, Masab Tank, Hyderabad, India in 2003, B.E. degree in Electronics and Communications Engineering from CBIT, Hyderabad, India in 2006 and M.Tech degree from Solid State Devices and VLSI group, Department of Electronics and Computer Science, Indian Institute of Technology Roorkee, Roorkee, India in 2008 under the joint supervision of Prof R.C.Joshi and Prof A.K.Saxena on "FPGA based parallel implementation of DWT for image compression"He is presently working as R&D Engineer at Teja's Networks.His research interests include Efficient implementation of wavelet algorithms, VLSI signal processing, Digital and Analog VLSI.