World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:9, 2008

VLSI Design of 2-D Discrete Wavelet Transform for Area-Efficient and High-Speed Image Computing

Mountassar Maamoun, Mehdi Neggazi, Abdelhamid Meraghni, and Daoud Berkani

Abstract**—**This paper presents a VLSI design approach of a high-speed and real-time 2-D Discrete Wavelet Transform computing. The proposed architecture, based on new and fast convolution approach, reduces the hardware complexity in addition to reduce the critical path to the multiplier delay. Furthermore, an advanced two-dimensional (2-D) discrete wavelet transform (DWT) implementation, with an efficient memory area, is designed to produce one output in every clock cycle. As a result, a very high-speed is attained. The system is verified, using JPEG2000 coefficients filters, on Xilinx Virtex-II Field Programmable Gate Array (FPGA) device without accessing any external memory. The resulting computing rate is up to 270 M samples/s and the (9,7) 2-D wavelet filter uses only 18 kb of memory (16 kb of first-in-first-out memory) with 256×256 image size. In this way, the developed design requests reduced memory and provide very high-speed processing as well as high PSNR quality.

Keywords**—**Discrete Wavelet Transform (DWT), Fast Convolution**,** FPGA, VLSI.

I. Introduction

THE Discrete Wavelet Transform (DWT) has become one of the most used techniques for image compression and is applied in a large category of applications [1]. DWT can provide significant compression ratios without great loss of visual quality than the previous techniques such as the Discrete Cosine Transform (DCT) and the Discrete Fourier Transform (DFT). The DWT present the main part of the JPEG2000 standard, which permits both lossy and lossless compression of digital images. It allows an encoded image to be reconstructed progressively.

The compression phase is mainly divided into three sequential steps: (1) Discrete Wavelet Transform, (2) Quantization, and (3) Entropy Encoding. After preprocessing, each component is independently analyzed by an appropriate discrete wavelet transform. The computational block diagram of the functionalities of the compression system is shown in Fig. 1.

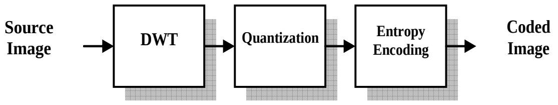

Fig. 1 DWT based compression system

Conventionally, programmable DSP chips are used to implement such algorithms for low-rate applications and the VLSI application specific integrated circuits (ASICs) for higher rates [2]. The FPGAs are programmable logic devices that provide sufficient quantities of logic resources that can be adapted to support a large parallel distributed architecture.

Lifting and convolution present the two computing approaches to achieve the discrete wavelet transform [3]. While conventional lifting-based architectures require fewer arithmetic operations compared to the convolution-based approach for DWT, they sometimes have long critical paths. If Ta and Tm are the delays of the adder and multiplier, respectively, then the critical path of the lifting-based architecture for the (9,7) filter is (4×Tm + 8×Ta), while that of the convolution implementation is (Tm + 2×Ta) [4]. In addition to this and for the reason to preserve proper precision, intermediate variables widths are larger in lifting-based computing. As a result, the lifting multiplier and adder delays are longer than the convolution ones.

Many VLSI lifting-based DWT architectures have been developed and implemented to reduce the memory requirements and the critical path [5]-[14]. A JPEG2000 single-chip encoder with 81 M samples/s is presented in [8]. A high-speed and area-efficient system is presented in [11], 7.5 Kb is used for the (5,3) wavelet filter and 30 Kb for the (9,7) one, with 256×256 image size. Recently, the new lifting-based architectures represent the faster option, and the main critical path is about Tlm, which represents the lifting multiplier delay.

The present work introduces new reflection of VLSI design for very high-speed image computing using new convolution-based DWT.

Firstly, new and fast convolution-based architecture is developed to reduce the hardware requirement complexity in addition to reduce the critical path to the multiplier delay (Tcm). Thus, we reach a very high-speed computing compared to the lifting option.

Secondly, a novel and adapted two-dimensional (2-D) discrete wavelet transform system, with an efficient memory area (e.g. 16 Kb for the (9,7) wavelet filter with 256×256 image size), is designed to produce one output in every clock cycle.

II. Discrete Wavelet Transform

A. One-Dimensional Discrete Wavelet Transform

The basic DWT can be realized by convolution-based implementation using the FIR-filters to do the transform. The input discrete signal X(n) is filtered by a low-pass filter (***h***) and a high-pass filter (***g***) at each transform level. The two output streams are then sub-sampled by simply dropping the alternate output samples in each stream to produce the low-pass subband ***Y_L*** and high-pass subband ***Y_H***. The associated equations can be written (1).

$$y_L(n) = \sum_{i=0}^{\frac{N}{2}-1} h(2n-i)\cdot x(i)\text{, } y_H(n) = \sum_{i=0}^{\frac{N}{2}-1} g(2n-i)\cdot x(i) \qquad (1)$$

Fig. 2 shows the signal analysis and reconstruction in one-dimensional (1-D) Discrete Wavelet Transform.

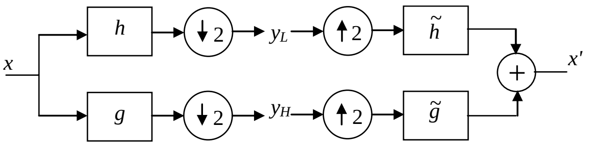

Fig. 2 Signal analysis and reconstruction in DWT

B. Two-Dimensional Discrete Wavelet Transform

For two-dimensional (Image) analysis and reconstruction the multi-resolution approach for Discrete Wavelet decomposition of signals using a pyramidal filter structure proposed by Mallat can be adopted. Fig. 3 shows the two-level multi-resolution wavelet decomposition of signals using pyramidal filter structure.

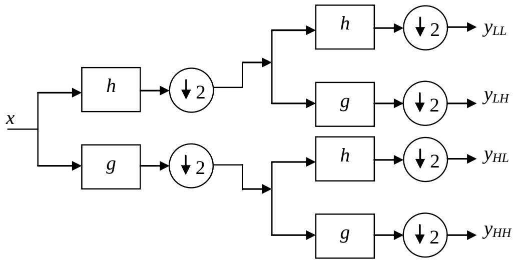

Fig. 3 Two-level multi-resolution wavelet decomposition

III. Fast Convolution-Based Discrete Wavelet Transform Architecture

There are many implementations of the convolution-based DWT [15]-[18]. A semi-systolic form of a VLSI architecture has been proposed by Acharya and Chen [15]. The proposed architecture, based on new and fast convolution approach, presents an implementation of a very high-speed discrete wavelet transform with reduced hardware complexity and memory. The main principle of the architecture can be applied to implement any symmetric filter. The (9, 7) wavelet filter presents the developed example. These (9, 7) filter has 9 low-pass filter coefficients ***h = { h-4 , h-3 , h-2 , h-1, ho, h1 , h2 , h3, h4}*** and *7* high-pass filter coefficients ***g = { g-2 , g-1, go, g1, g2 , g3 , g4}***.

Based on the low-pass filter coefficients symmetry (***h-i = hi)***, the equation 2 can be given.

$$\begin{aligned}
y_{L0} &= h_0(0+x_0)+h_1(0+x_1)+h_2(0+x_2)+h_3(0+x_3)+h_4(0+x_4)\\
y_{L1} &= h_0(0+x_2)+h_1(x_1+x_3)+h_2(x_0+x_4)+h_3(0+x_5)+h_4(0+x_6)\\
y_{L2} &= h_0(0+x_4)+h_1(x_3+x_5)+h_2(x_2+x_6)+h_3(x_1+x_7)+h_4(x_0+x_8)\\
&\vdots
\end{aligned} \qquad (2)$$

$$y_{L\frac{N}{2}-2} = h_0(x_{N-4}+0)+h_1(x_{N-5}+x_{N-3})+h_2(x_{N-6}+x_{N-2})+h_3(x_{N-7}+x_{N-1})$$
$$+h_4(x_{N-8}+0)$$
$$y_{L\frac{N}{2}-1} = h_0(x_{N-2}+0)+h_1(x_{N-3}+x_{N-1})+h_2(x_{N-4}+0)+h_3(x_{N-5}+0)+$$
$$h_4(x_{N-6}+0)$$

Similarly, the high-pass filter coefficients present symmetry as follows:

$$\begin{aligned}
g_{-2} &= \tilde{h}_3 = g_4,\\
g_{-1} &= \tilde{h}_2 = g_3,\\
g_0 &= \tilde{h}_1 = g_2,\\
g_1 &= \tilde{h}_0.
\end{aligned} \qquad (3)$$

As a result (4) can be given.

$$\begin{aligned}
y_{H0} &= g_1(0+0)+g_2(0+x_0)+g_3(0+x_1)+g_4(0+x_2)\\
y_{H1} &= g_1(0+x_1)+g_2(x_0+x_2)+g_3(0+x_3)+g_4(0+x_4)\\
y_{H2} &= g_1(0+x_3)+g_2(x_2+x_4)+g_3(x_1+x_5)+g_4(x_0+x_6)\\
&\vdots
\end{aligned} \qquad (4)$$

$$y_{H\frac{N}{2}-2} = g_1(0+x_{N-5})+g_2(x_{N-6}+x_{N-4})+g_3(x_{N-7}+x_{N-3})+g_4(x_{N-8}+x_{N-2})$$
$$y_{H\frac{N}{2}-1} = g_1(0+x_{N-3})+g_2(x_{N-4}+x_{N-2})+g_3(x_{N-5}+x_{N-1})+g_4(x_{N-6}+0)$$

The architecture to compute the ***Y_Li*** and ***Y_Hi*** is shown in Fig. 4. Additional registers are added, between multipliers and adders, to speed up the computing. The critical path is reduced to the multiplier delay (Tcm). Furthermore, the outputs ***Y_Li*** and ***Y_Hi*** are obtained alternately at the trailing edges of the even and odd clock cycles. (e.g., ***Y_L0, Y_L1, Y_L2, …..*** are obtained at clock cycles 9, 11, 13, . . . and ***Y_H0, Y_H1, Y_H2, …*** are obtained at clock cycles 8, 10, 12, .. . respectively).

The preliminary gate count estimates and number of components used in the proposed fast convolution-based architecture are given in Table I.

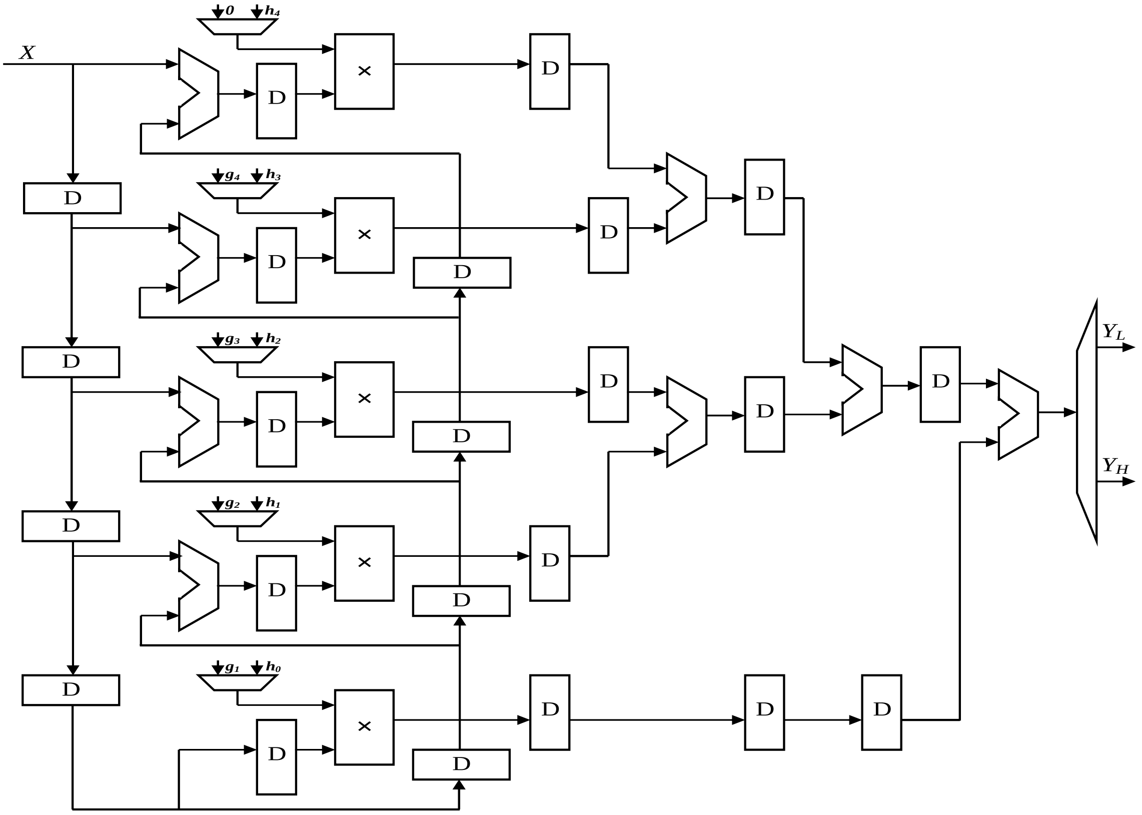

Fig. 4 The proposed fast convolution-based 1-D (9,7) DWT block diagram

TABLE I
PRELIMINARY GATE COUNT ESTIMATES AND NUMBER OF COMPONENTS USED IN THE PROPOSED ARCHITECTURE

Component	Gate count	Number of units
Adder (8b+8b)	200	4
Adder (16b+16b)	400	4
Register (8 bit)	50	8
Register (9 bit)	56	5
Register (16 bit)	100	10
Multiplier (8b x 9b)	810	5

The total number of the used gate and the estimated work frequency for the (9,7) lifting architecture and the proposed one, using the field programmable gate array Xilinx Virtex-II, is provided in Table II.

TABLE II
THE TOTAL NUMBER OF THE USED GATE AND THE ESTIMATED WORK FREQUENCY

Architecture	The total gate	Work frequency
The lifting architecture [6]	13400	210 Mhz
The proposed architecture	8130	270 Mhz 275 Mhz*

*With 7 bits coding coefficients

IV. THE PROPOSED FAST CONVOLUTION-BASED 2-D DWT ARCHITECTURE

In this approach and to avoid the need of a transpose circuit between the two levels, the system stars the column-processing as soon as sufficient numbers of rows have been filtered. Fig. 5 presents the main 2-D DWT fast convolution-based diagram.

The proposed FPGA implementation of the 2-D Discrete Wavelet Transform is designed with two fast convolution-based blocks. The first one, which is similar to the 1-D block, realizes the row Discrete Wavelet Transform and uses D-Latch devices for the X(n) storage. The second achieves the Column Discrete Wavelet Transform using an FPGA block RAM storage of the computed rows.

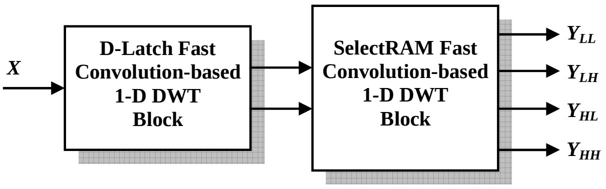
X
D-Latch Fast Convolution-based 1-D DWT Block
SelectRAM Fast Convolution-based 1-D DWT Block
YLL
YLH
YHL
YHH

Fig. 5 2-D DWT fast convolution-based block diagram

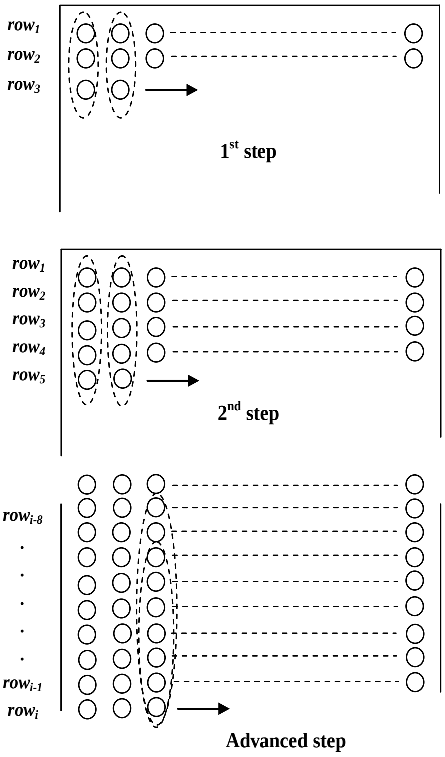

Fig. 6 The SelectRAM based fast convolution computing

$\boldsymbol{Y_L}$ and $\boldsymbol{Y_H}$ present the first level outputs. The $\boldsymbol{Y_{LL}}$, $\boldsymbol{Y_{LH}}$, $\boldsymbol{Y_{HL}}$ and $\boldsymbol{Y_{HH}}$ present the second level and the 2-D DWT outputs. $\boldsymbol{Y_{L0}}$, $\boldsymbol{Y_{L1}}$, $\boldsymbol{Y_{L2}}$, ….. are obtained at clock cycles 9, 11, 13, … and $\boldsymbol{Y_{H0}}$, $\boldsymbol{Y_{H1}}$, $\boldsymbol{Y_{H2}}$, … are obtained at clock cycles 8, 10, 12, .. . respectively.

The first row of $\boldsymbol{Y_{LH}}$ and $\boldsymbol{Y_{HH}}$ can be obtained after the beginning of the third row storage of the first level outputs. After the beginning of the fifth storage of the first level outputs, we can obtain the second row of $\boldsymbol{Y_{LH}}$ and $\boldsymbol{Y_{HH}}$ and the first row of $\boldsymbol{Y_{LL}}$ and $\boldsymbol{Y_{HL}}$ (Fig. 6).

Nine FPGA block RAM in Dual-Port Mode are required to accomplish the second level of the Parallel Distributed 2-D DWT Architecture with the (9,7) wavelet filters. The SelectRAM 1-D DWT fast convolution-based Modules is composed by blocks RAM, interface circuit and a computing circuit (Fig. 7). The computing circuit presents the main architecture of the D-Latch 1-D DWT fast convolution-based.

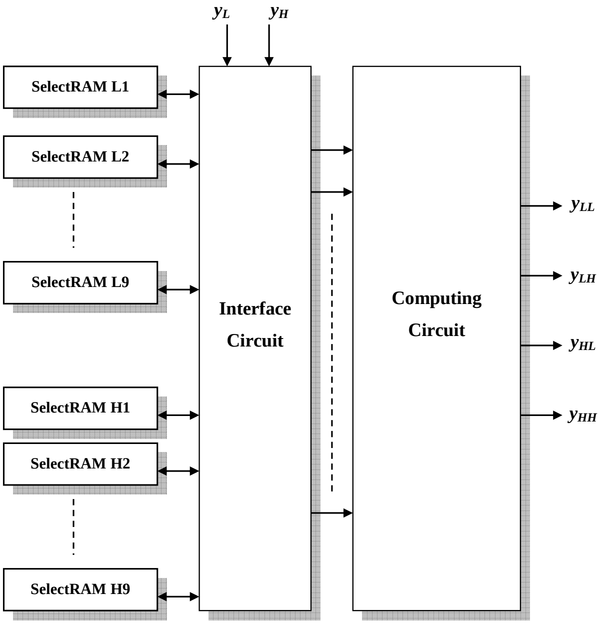

Fig. 7 The SelectRAM 1-D DWT fast convolution-based block diagram

At each step of computing, only one block RAM is selected in write mode:

1st step:
Block RAM 1 and block RAM 2: read mode
Block RAM 3: write mode

2nd step:
Block RAM 1 to block RAM 4: read mode
Block RAM 5: write mode

Advanced step:
Block RAM 1 to block RAM8: read mode
Block RAM 9: write mode

We have implemented the 2-D DWT Parallel Distributed Architecture described in the previous section using one of the XC2V1000 Xilinx Virtex-II FPGA devices. This device contains 1M system gates (5,120 slices), 40 Multiplier Blocks and can operate at a maximum clock speed of 450 MHz [19].

Each Xilinx Virtex-II FPGA block SelectRAM cell is an 18 Kbit fully synchronous memory. The two ports have independent inputs and outputs and are independently clocked. The data widths of the two ports can be configured independently, providing built-in bus-width conversion. As a result, one output is produced in every clock cycle and this configuration is able to accomplish 2048×2048 block computing.

Using the first-in-first-out (FIFO) memory, the total memory size (in words) needed 2_D_M, for the proposed one level 2-D computing, is

$$2\_D\_M = (K-1) \times N \tag{5}$$

We assume that: (1) The maximum number of filter taps among the high-pas and low-pass filters is K; (2) the image size is N×N. Fig. 8 presents the main FIFO-based 2-D DWT fast convolution-based diagram.

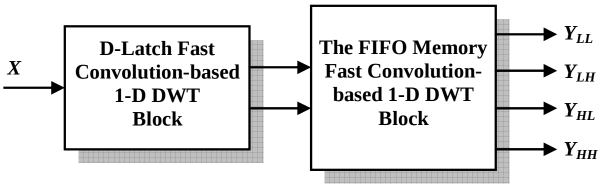
X
D-Latch Fast Convolution-based 1-D DWT Block
The FIFO Memory Fast Convolution-based 1-D DWT Block
Y_{LL}
Y_{LH}
Y_{HL}
Y_{HH}

Fig. 8 The FIFO-based 2-D DWT fast convolution-based block diagram

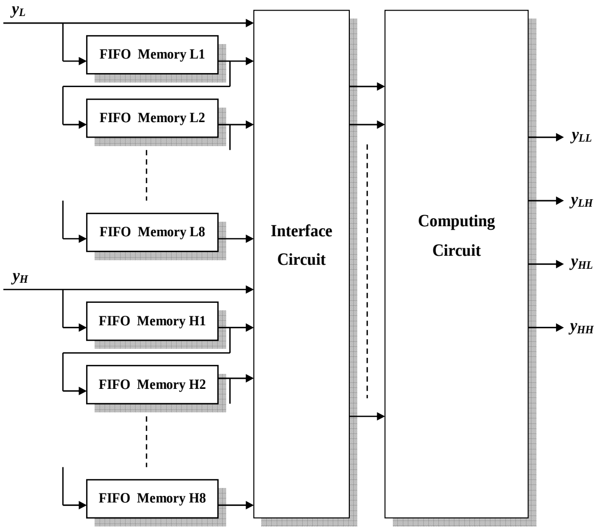

Fig. 9 The FIFO Memory 1-D DWT fast convolution-based block diagram

According to Eq. (5), only eight blocks in Dual-Port are needed to compute the (9,7) wavelet filter and four blocks in Dual-Port are for the (5,3) one. Fig. 9 shows the (9,7) FIFO Memory 1-D DWT fast convolution-based block diagram.

At each step of computing, all Memory blocks are selected in FIFO mode:

1st step:
Memory block 1 and memory block 2: FIFO mode

2nd step:
Memory block 1 to memory block 4: FIFO mode

Advanced step:
Memory block 1 to memory block 8: FIFO mode

We conclude that the fast convolution-based scheme and the appropriate 2-D DWT architecture reduce the memory area and the critical path to the convolution multiplier delay, which is the smallest one. The memory resources comparisons, with 256×256 image size, between our 2-D architecture and others are listed in Table III.

TABLE III
MEMORY RESOURCE IN 2-D DWT ARCHITECTURE

Architecture	**Memory Resource** **(Sing or Dual Port for one-level)**	**Computing speed**
Proposed (Image size: 256×256)	Very Reduced Memory 8 Kb for (5,3) and 16 Kb for (9,7)	Very High-Speed
[18] (Image size: 256×256)	Very Reduced Memory 10 Kb for (5,3) and 18 Kb for (9,7)	High-Speed
[11] (Image size: 256×256)	Very Reduced Memory 7,5 Kb for (5,3) and 30 Kb for (9,7)	High-Speed
[6][7][8][9][14] (Image size: 256×256)	Reduced Memory	High-Speed

The PSNR, for the selected images, after forward and inverse Discrete Wavelet Transform with 7 bits coding coefficients, are given in Table IV.

TABLE IV
PSNR VALUES AFTER FORWARD AND INVERSE DISCRETE WAVELET TRANSFORM

Image	Gold Hill	Lena	Barabara	Baboon	Peppers
PSNR	91,89	91,36	91,03	90,93	89,80

V. CONCLUSION

In this paper, we have proposed a parallel architecture for very high-speed computing Discrete Wavelet Transform using SRAM and FIFO memory. To produce one output in every clock cycle in addition to reduce the critical path as well as an efficient memory area, new fast convolution-based architecture approach is performed. In this approach, the system stars the column-processing as soon as sufficient numbers of rows have been filtered. Two fast convolution-based blocks, for the two-dimensional (2-D) discrete wavelet transform (DWT), are used to accelerate the computing.

The Xilinx Virtex-II FPGA implementation, of developed architecture using the (9,7) filter, required eight FIFO memory in Dual-Port Mode. The (3,5) 2-D wavelet filter uses only 8 kb of memory with 256×256 image size and the (9,7) one uses only 16 kb.

REFERENCES

[1] Mallat, S.: A Theory for Multiresolution Signal Decomposition: The Wavelet Representation. IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 11, No. 7. (1989) 674-693.

[2] Gnavi, S., Penna, B., Grangetto, M., Magli, E., Olmo, G.: Wavelet kernels on a DSP: A comparison between lifting and filter banks for image coding. Applied Signal Processing: Special Issue on Implementation of DSP and Communication Systems. Vol. 2002. No. 9. (2002) 981-989.

[3] Daubechies, I., Sweldens, W.: Factoring wavelet transforms into lifting schemes. The Journal of Fourier Analysis and Applications. vol. 4. (1998) 247-269.

[4] Acharya, T., Tsai, P.S.: JPEG2000 Standard for Image Compression. A John Wiley & Sons, Inc. USA (2005).

[5] Andra, K., Chakrabarti, C, Acharya,T.: A VLSI Architecture for Lifting-Based Forward and Inverse Wavelet Transform. IEEE Transactions on Signal Processing, vol. 50. No. 4. (2002) 966-977.

[6] Andra, K., Acharya,T., Chakrabarti, C.: A High- Performance JPEG2000 Architecture. IEEE Transactions on Circuits and Systems for Video Technology, vol. 13. No. 3. (2003) 209-218.

[7] B.-F.Wu and C.-F. Lin, "An efficient architecture for JPEG2000 coprocessor," IEEE Trans. Consum. Electron., vol. 50, no. 4, pp. 1183–1189, Nov. 2004.

[8] H.-C. Fang, C.-T. Huang, Y.-W. Chang, T.-C.Wang, P.-C. Tseng, C.-J. Lian, and L.-G. Chen, "81 MS/s JPEG2000 single-chip encoder with rate-distortion optimization," in Proc. ISSCC Tech. Dig., 2004, vol. 1, pp. 28–531.

[9] Q. P. Huang, R. Z. Zhou, and Z. L. Hong, "Low memory and low complexity VLSI implementation of JPEG2000 codec," IEEE Trans. Consum. Electron., vol. 50, no. 2, pp. 638–646, May 2004.

[10] Descampe, A, et al: A Flexible, Hardware JPEG 2000 Decoder for Digital Cinema. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 16, No 11. (2006) 1397-1410.

[11] K. Z. Mei, N. N. Zheng, C. Huang, Y. Liu, and Q. Zeng "VLSI Design of a High-Speed and Area-Efficient JPEG2000 Encoder," IEEE Trans. Circuits Syst. Video Technol., vol. 17, no. 8, pp. 1065–1078, Agu. 2007.

[12] JPEG2000 Decoder: BA109JPEG2000D Factsheet.Barco-Silex. (2008).

[13] JPEG 2000 Video CODEC (ADV212). Analog Devices. (2008).

[14] CS6510 JPEG2000 Encoder Amphion Inc. [Online]. Available: http://www.amphion.com

[15] Acharya, T., Chen, P.: VLSI Implementation of a DWT Architecture. Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS). Monterey, CA. (1998).

[16] Acharya, T.: Architecture for Computing a Two-Dimensional Discrete Wavelet Transform. US Patent 6178269. (2001).

[17] N.D. Zervas, G.P. Anagnostopoulos, V. Spiliotopoulos, Y. Andreopoulos, C.E. Goutis, "Evaluation of design alternatives for the 2-D-discrete wavelet transform," IEEE Trans. Circuits Syst. Video Technol., Vol.11, no. 12, pp. 1246–1262. Dec. 2001.

[18] G. Dimitroulakos, M.D. Galanis, A. Milidonis, and C.E. Goutis, "A high-throughput, memory efficient architecture for computing the tile-based 2D discrete wavelet transform for the JPEG2000," INTEGRATION, the VLSI journal., vol. 39, no. 1, pp. 1-11, 2005.

[19] VirtexTM-II platform FPGAs: Complete Data Sheet. Xilinx. (2007).