Indian Journal of Communications Technology and Electronics (IJCTE)
Vol.2.No.1 2014pp 54-59
available at: www.goniv.com
Paper Received :05-03-2014
Paper Published:28-03-2014
Paper Reviewed by: 1. John Arhter 2. Hendry Goyal
Editor : Prof. P.Muthukumar

FPGA IMPLEMENTATION OF MEMORY EFFICIENT HIGH SPEED STRUCTURE FOR MULTILEVEL 2D-DWT

Mr.SenthilKumar.K, M.E.,
Assistant professor, Dept. of ECE, DMI College of Engineering, Palanchur.
maksen79@gmail.com

Ms.LeemaRose.D, B.E.,
Student,Dept. of ECE, DMI College of Engineering, Palanchur.
leemarose1122@gmail.com

ABSTRACT

In the imaging field compression of images is very much needed for the efficient transmission of the image. 2D-DWT is widely used in achieving the compression of images. The objective of my project is to propose memory efficient high speed architecture for multilevel 2D – DWT using FPGA. The system is said to be high speed since it uses parallel and pipelined processing of the image. The proposed architecture is based on the Lifting based DWT scheme. This scheme effectively reduces the hardware complexity and memory accesses. The proposed structure offers significant saving of area and power due to the substantial reduction in usage of memory. The design is realised in VHDL language and optimized in terms of throughput and memory requirements. The implementations are completely parameterized with respect to the size of the input image and the number of decomposition levels. From the result obtained, the proposed architecture used less number of components from the available components with less memory usage of 198 KB

Keywords – 2D- Discrete Wavelet Transform (DWT), lifting scheme

1. INTRODUCTION

The discrete wavelet transform (DWT) has gained wide popularity due to its excellent de-correlation property. Many modern image and video compression systems embody the DWT as the transform stage. One of the prominent features of JPEG2000 standard, providing it the resolution scalability, is the use of the two dimensional Discrete Wavelet Transform (2D-DWT) to convert the image samples into a more compressible form. It is considered as the key difference between JPEG and JPEG2000 standards. The features of DWT are it allows image multi resolution representation in a natural way because more wavelet sub bands are used to progressively enlarge the low frequency sub bands.It supports wavelet coefficients analysis in both space and frequency domains, thus the interpretation of the coefficients is not constrained to its frequency behaviour and can perform better analysis for image vision and segmentation.For natural images, the DWT achieves high compactness of energy in the lower frequency sub bands, which is extremely useful in applications such as image compression.

2. ONE DIMENSIONAL DISCRETE WAVELET TRANSFORM

Two main methods exist for the implementation of 1DDWT: the traditional convolution-based implementation and the lifting-based implementation.

A. Convolution Based DWT

In the traditional implementation of DWT, a pair of finite impulse response filters (FIR) is applied in parallel, highpass and low-pass filter. Each filtering operation is shown in Figure1.

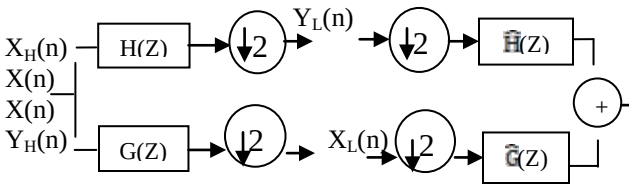

Figure 1 Single 1D DWT block

The input sequence X(n) in Figure 1 is convolved with the quadrature mirror filters H(z) and G(z) and the outputs obtained are decimated by a factor of two. After downsampling, alternate samples of the output sequence from the low pass filter and high pass filter are dropped. This reduces the time resolution by half and conversely doubles the frequency resolution by two. The 1D-DWT is a two channel sub-band decomposition of an input signal X(n) that produces two subband coefficients $Y_L(n)$ and $Y_H(n)$ for one-stage of decomposition according to the following equations

$$Y_L(n) = \sum_{i=0}^{\tau L-1} H(i)x(2n-1) \quad (1)$$

$$Y_H(n) = \sum_{i=0}^{\tau H-1} G(i)x(2n-1) \quad (2)$$

In the synthesis stage, scaling and wavelet coefficients $Y_L(n)$ and $Y_H(n)$ are treated inversely by up-sampling and filtering with low pass $\hat{H}(z)$ and high pass $\hat{G}(z)$ filters to perform reconstruction. This stage is also called Inverse Discrete Wavelet Transform (IDWT). Original and reconstructed signals are generally different, unless the two filters H and G satisfy some relationships. The perfect reconstruction condition consists in ensuring no distortion and no aliasing of the reconstructed data.

B. Lifting based DWT

The convolution-based 1-D DWT requires both a large number of arithmetic computations and a large memory for storage. Such features are not desirable for either high speed or low-power image processing applications. Recently, a new mathematical formulation for wavelet transformation has been proposed by Swelden as a light-weighted computation method for performing wavelet transform. The main feature of the lifting-based wavelet transform is to break-up the high pass and the low pass wavelet filters into a sequence of smaller filters. The lifting scheme requires fewer computations compared to the convolution-based DWT. Therefore the computational complexity is reduced to almost a half of those needed with a convolution approach. As a result, lifting has been suggested for implementation of DWT in JPEG2000 standard. The lifting-based wavelet transform basically consists of three steps, which are called split, lifting, and scaling, respectively, as shown in Figure 2.

The basic idea of lifting scheme is first to compute a trivial wavelet (or lazy wavelet transform) by splitting the original 1-D signal into odd and even indexed subsequences, and then modifying these values using alternating prediction and updating steps.

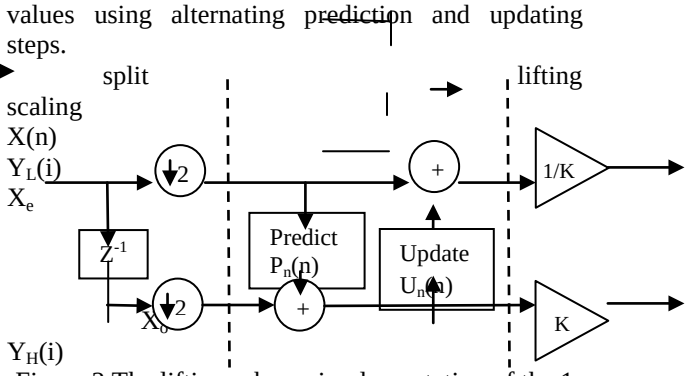

Figure 2 The lifting scheme implementation of the 1 D- DWT

The lifting scheme algorithm can be described as follow:

Split step*:* The original signal, *X(n)*, is split into odd and even samples (lazy wavelet transform).

Lifting step*:* This step is executed as N sub-steps (depending on the type of the filter), where the odd and even samples are filtered by the prediction and update filters, *Pn(n)* and *Un(n)*.

Normalization *or* ***Scaling step****:* After N lifting steps, a scaling coefficients K and 1/K are applied respectively to the odd and even samples in order to obtain the lowpass band ($Y_L(i)$), and the high-pass sub-band ($Y_H(i)$).

The table 1 shows the number of multiplications, additions and shifts needed for both methods. Simulation was performed by using Lena image (512x512).

TABLE 1
COMPLEXITY COMPARISON OF CONVOLUTION AND LIFTING BASED IMPLEMENTATION

		Multiplication	**Addition**	**Shift**
(9,7) filter	**Convolution**	3670216	4194504	None
	Lifting	1579108	2109540	None
(5,3) filter	**Convolution**	490	2890	3340
	Lifting	None	1940	1420

Thus from the above table 1 convolution-based DWT is computationally extensive and resulting to be area, power, and memory hungry. Lifting scheme reduces the computations up to 50%, which affect directly the memory, surface and the power consumption of the system. To sum up, lifting scheme will be more suitable for hardware implementation with limited on-ship memory, lower computational complexity, small area and low power.

3. TWO-DIMENSIONAL DISCRETE WAVELET TRANSFORM

The basic idea of 2-D architecture is similar to 1-D architecture. A 2-D DWT can be seen as a 1-D wavelet transformalong the rows

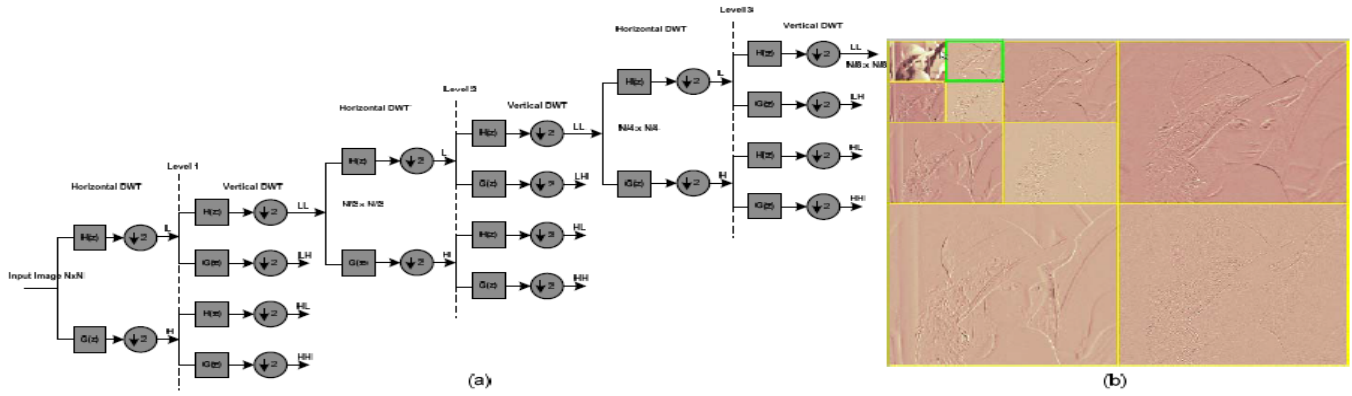

Figure 3 Three-level decomposition algorithm for 2D-DWT and Lena image decomposition

Then a 1-D wavelet transform along the columns. The 2-D DWT operates in a straightforward manner by inserting array transposition between the two 1-D DWT. The rows of the array are processed first with only one level of decomposition. This essentially divides the array into two vertical halves, with the first half storing the average coefficients, while the second vertical half stores the detail coefficients. This process is repeated again with the columns, resulting in four sub-bands (see Figure 1.4a) within the array defined by filter output. Figure 3 shows a three-level 2-D DWT decomposition of the Lena image. The LL sub-band represents an approximation of the original image, the LL1 sub-band can be considered as a 2:1 sub-sampled (both horizontally and vertically) version of the original image. The other three sub-bands HL1, LH1, and HH1 contain higher frequency detail information (mostly local discontinuities in the edges of the image).This process is repeated for as many levels of decomposition as are desired.

4. ARCHITECTURE

The proposed architecture is a pipeline and memory efficient based on lifting scheme. It improves the implementation of the 2D-DWT by adopting an efficient usage of hardware resources, low control complexity; reducing the embedded memory requirements and external memory access. We exploit the data dependency of the lifting scheme technique and propose a new design of 1D-DWT architecture. The key idea consists of pipelining and interleaving the operations between row and column processing to increase throughput and reduce latency. The architecture minimizes the number of external memory access, reduces the power consumption and employs small embedded memory for intermediate data storage.

The input signals in 1-D DWT are transformed into a high (H) and a low (L) component signals by a 1-D filter bank. Four lifting steps are required to compute the high pass coefficients and the low pass coefficients. Based on the operation of the row processing the straightforward implementation is to apply the first step of the Lifting Calculation (LC1) to all the input samples and store the transformed coefficients, and then apply the second step (LC2). The same process is repeated for the third and fourth steps. However, this approach requires high embedded memory usage (to store intermediates coefficients), large amount of access to external memory (to access the even index input samples for LC2), and will results in huge latency. To overcome all these issues, we propose to start the computation of LC2, LC3 and LC4 as soon as enough data are available (2 coefficients are produced) in order to reduce the LC2, LC3 and LC4 latency. The register block is also used between each processor to locally store the intermediate results computed by the previous step and the even data.

The 1D-DWT block reads two inputs data in one clock cycle, because the even and odd image data are stored in one memory case. Figure 3.4 shows the internal architecture of 1D-DWT block. It consists of four instances of the Lifting Calculation blocks LC1, LC2, LC3 and LC4 and the intermediate registers blocks denoted by Register_1, Register_2, Register_3 and Register_4 to store the intermediate data needed between the lifting steps.

In general, all the lifting steps are essentially in the form:

$$Y_i = X_i + a(X_{i-1} = X_{i+1}) \quad (3)$$

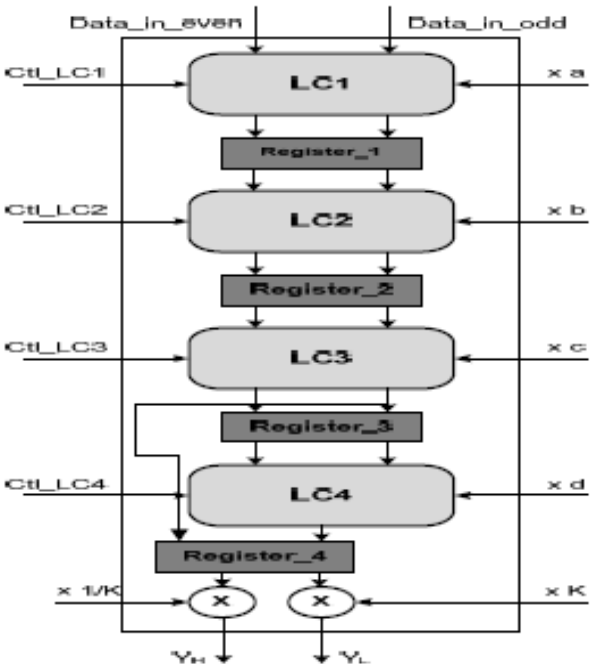

Figure 4 The block diagram of 1D-DWT architecture
So we need a different configuration of adders, and multipliers that are connected in a manner that will support the computational structure of the lifting steps.

6. BLOCK DIAGRAM

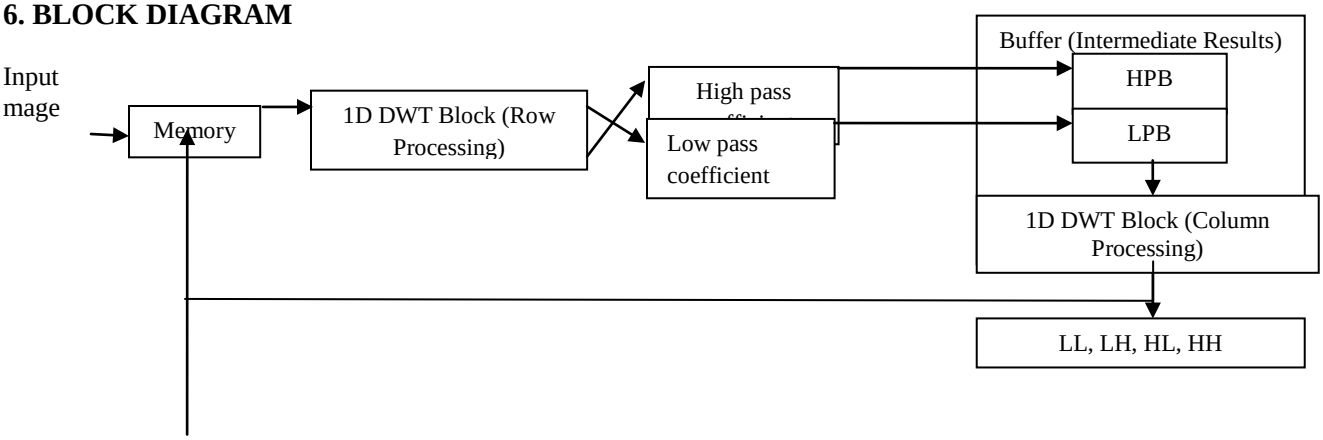

Figure 5.Simplified block diagram of 2D DWT

7. ALGORITHM

Step 1: The input image is given as bitmap image to the data converter
Step 2: The data of the image is given as matrix format to the 1D DWT block for row processing
Step 3: After the row processing, the image pixels are divided into high pass and low pass coefficient
Step 4: The intermediate results from row processing is stored in a buffer
Step 5: The resultant image data from previous step is given to another 1D DWT block for column processing
Step 6: Due to column processing, the image is further divided into four sub bands as LL, LH,HL, HH.
Step 7: For next level DWT, the LL sub band from previous step is taken and again the procedure is repeated from step 1.
Step 8: Finally the hardware utilization of the above DWT process is tabulated.

8. RESULT

The VHDL coding for performing 2D-DWT is executed in Xilinx ISE. From the result obtained, the proposed structure used less number of

components from the available components in order to perform 2D- DWT in an efficient manner. Thus the architecture reduces the hardware utilization to 25% due to lifting based DWT scheme. And also results in saving of area and power due to

substantial reduction in the usage of memory. In this project two levels of decomposition has been carried out in 2D- DWT with the memory usage of 198 KB. The RTL schematic is shown in Figure 6 and the top level implementation of the hardware module is shown in Figure 7. The summary of total hardware usage from the available components is given in the table 2.

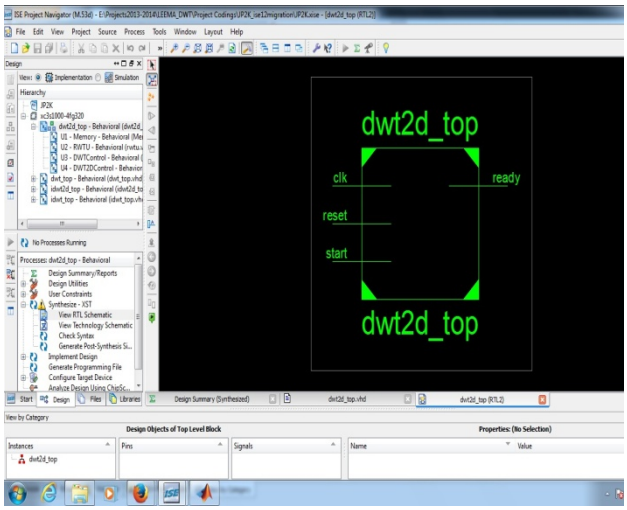

Figure 6. RTL schematic of 2D-DWT

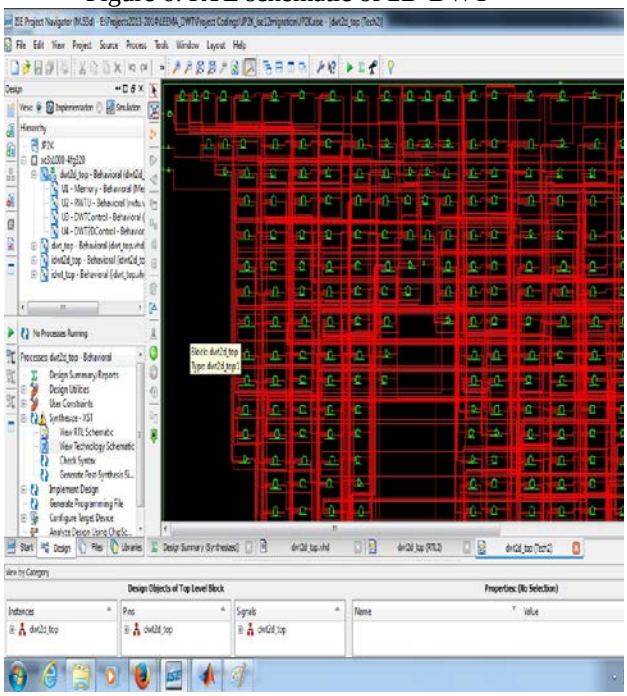

Figure 7 Top level implementation of hardware module

Table 2 Summary of total hardware usage from available components

DEVICE UTILIZATION SUMMARY (ESTIMATED VALUES)			
LOGIC UTILIZATION	**USED**	**AVAILABLE**	**UTILIZATION**
No. of slices	420	7680	5%
No. of slice flip flos	309	15360	2%
No. of 4 input LUTs	797	15360	5%
No. of bonded IOBs	4	221	1%
No. of GCKs	1	8	12%

9. CONCLUSION

Thus a memory efficient high speed structure for multilevel 2D-DWT is proposed in this project to meet the requirement of image compression. The advantages of the proposed architecture are saving embedded memories, fast computing time, low power consumption and low control complexity. The proposed architecture has been correctly verified by VHDL Language. The future work of this project is hardware implementation of the proposed architecture using SPARTAN 3E.

REFERENCES

[1] P.Arulselvan, C.Karthik, M.Peer Mohamed, "FPGA Implementation of Efficient Fast Convolution Architecture Based Discrete Wavelet Transform" International Journal of Engineering Research & Technology (IJERT)ISSN: 2278-0181IJERTV2IS90644 www.ijert.orgVol. 2 Issue 9, September – 2013

[2] Basant Kumar Mohanty, *Senior Member, IEEE,* and Pramod Kumar Meher, *Senior Member, IEEE,* "Memory-Efficient High-Speed Convolution-based Generic Structure for Multilevel 2-D DWT"IEEE Transactions on circuits and systems for video technology, vol. 23, no. 2, February 2013.

[3] Z. Gao and C. Xiong, "High-throughput implementation of lifting-based discrete wavelet transform using look-ahead pipelining," *SPIE, Opt.Eng.*, vol. 49, no. 10, 107003, pp. 1–7, Oct. 2010.

[4] C.-H. Hsia, J.-M.Guo, and J.-S. Chiang, "Improved low-complexity algorithm for 2-D integer lifting-based discrete wavelet transform using symmectric mask-based scheme," *IEEE Trans.*

Circuit Syst. VideoTechnol., vol. 19, no. 8, pp. 1202–1208, Aug. 2009.

[5] Y.-K. Lai, L.-F.Chen, and Y.-C. Shih, "A high-performance and memory-efficient VLSI architecture with parallel scanning method for 2-D lifting-based discrete wavelet transform," *IEEE Trans. Consum.Electron.*,vol. 55, no. 2, pp. 400–407, May 2009.\

[6] A. Mansouri, A. Ahaitouf, and F. Abdi, "An Efficient VLSI Architecture and FPGA Implementation of High-Speed and Low Power 2-D DWT for (9, 7) Wavelet Filter" IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.3, March 2009

[7] P. K. Meher, B. K. Mohanty, and J. C. Patra, "Hardware-efficient systolic-like modular design for 2-D discrete wavelet transform," *IEEETrans Circuits Syst. II, Exp. Briefs*, vol. 55, no. 2, pp. 151–154, Feb. 2008.

[8] B. K. Mohanty and P. K. Meher, "Memory-efficient modular VLSI architecture for high-throughput and low-latency implementation of multilevel lifting 2-D DWT," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2072–2084, May 2011.

[9] M. Nibouche, A. Bouridane, F.Murtagh and O. Nibouche, "FPGA-Based Discrete Wavelet Transforms System" School of Computer Science, The Queen's University of Belfast. 18, Malone Road - Belfast BT7 1NN – UK

[10] M. Vishwanath, "The recursive pyramid algorithm for the discrete wavelet transform," *IEEE Trans. Signal Process.*, vol. 42, no. 3, pp. 673–676, Mar. 1994.