Design and VLSI Implementation of Efficient Discrete Wavelet Transfer Scheme.

N.Veera Durga[1], P.V.S.Muralidhar[2], G. Rajesh Kumar[3]

[1]P.G.Scholar, Dept. of ECE, Kakinada Institute of Engineering and Technology, A.P, India.
[2]Asst.Prof, Dept. of ECE, Kakinada Institute of Engineering and Technology, A.P, India.
[3]Asst.Prof, Dept. of ECE, SRKR College of Engineering, A.P, India.

Abstract - *This paper presents Design and VLSI implementation of efficient Discrete Wavelet Transfer scheme. In general the medical images need more accuracy without loosing accuracy. This concept is more effective than the Fourier transform and Short Time Fourier transform. The DWT was based on time-scale representation, which provides efficient multi-resolution. A major advantage of the proposed scheme is to reduce the complexity, large storage, reducing the number and period of clock cycles for the computation of wavelet transform. In proposed method, 5- stage pipelined architecture shares the partial load of the next stage with the present stage to reduce computational load at the next stage and critical path delay . The proposed architecture has replaced the multiplications by optimized shift and add operations to reduce the critical path delay . Implementation results show that the proposed architecture benefits from the features of reduced memory, low power consumption, low latency, and high throughput over several existing designs . The simulation results of DWT were verified with the appropriate test cases. Once the functional verification is done, discrete wavelet transform is implemented in VHDL and synthesized by using Xilinx ISE simulator tool in Virtex 5 FPGA family.*

***Key Words* - Discrete Wavelet Transform, pipelined scheme, Lifting-Based Scheme, critical path delay, Multi-Resolution, VLSI Architecture.**

1. INTRODUCTION

The fundamental idea of wavelet transforms is that the transformation should allow only changes in time extension, but not shape. This is effected by choosing suitable basic functions that allow for this changes. Changes in the time extension are expected to conform to the corresponding analysis frequency of the basis function. Wavelet compression is a one of the technique for image compression.. The goal is to store image data in as little space as possible in a file. Wavelet compression can be either lossless or lossy.[1] Using a wavelet transform, the wavelet compression methods are adequate for representing transients, such as percussion sounds in audio, or high-frequency components in two-dimensional images, for example an image of stars on a night sky. This means that the transient elements of a data signal can be represented by a smaller amount of information than would be the case if some other transform, such as the more widespread discrete cosine transform, had been used.

Discrete wavelet transform has been successfully applied for the compression of electrocardiograph (ECG) signals[2] In this work, the high correlation between the corresponding wavelet coefficients of signals of successive cardiac cycles is utilized employing linear prediction. Wavelet compression is not good for all kinds of data: transient signal characteristics mean good wavelet compression, while smooth, periodic signals are better compressed by other methods, particularly traditional harmonic compression. These coefficients can then be compressed more easily because the information is statistically concentrated in just a few coefficients. This principle is called transform coding. Still the image compression technique based on 2-D discrete wavelet transform (DWT) has recently gained superiority above traditional JPEG based upon discrete cosine transform and is standardized in varieties like JPEG2000 [5]. Quite similarly, the use of their 3-D superset, i.e. 3-D-DWT on video, beats the current predictive code standards, like H. 261-3, MPEG1-2, 4 by making the quality features just like better peak signal-to-noise rate (PSNR), removal of blocky artifacts in low bit rates. Furthermore, it features the added provisions of highly scalable compression, which is mostly coveted in modern communications over heterogeneous channels like the Net [6]. Successful program of 3-D-DWT has recently been reported inside the literary works in emerging fields just like medical image compression [6], hyper-spectral and space image compression [7], etc. Software-based techniques are played around with to combat the enormous computational complexity and memory space requirement associated with 3-D-DWT realization [8], [9]. Although the processor speed of modern computers soars excessive at the order of GHz, data fetching and communicating with external memories consume several T states, making the computation quite slower at the end.

2. LITERATURE SURVEY

2.1. Types of compressions

There are two types of compressions
1..Lossless compression:
Digitally identical to the original image. Only achieve a modest amount of compression
2. Lossy compression :
Discards components of the signal that are known to be redundant. Signal is therefore changed from input. Lossless compression involves with compressing data, when decompressed data will be an exact replica of the original data. This is the case when binary data such as executable are compressed.

2.2. Introduction to Wavelet Transform

The wavelet transform is computed separately for different segments of the time-domain signal at different frequencies. Multi-resolution analysis: analyzes the signal at different frequencies giving different resolutions. Good for signal having high frequency components for short durations and low frequency components for long duration, e.g. Images and video frames. A 'wavelet' is a small wave which has its energy concentrated in time. It has an oscillating wavelike characteristic but also has the ability to allow simultaneous time and frequency analysis and it is a suitable tool for transient, non-stationary or time-varying phenomena.
The wavelet theory has been studied by many researchers to answer the better and more appropriate functions to represent signals then the one offered by the Fourier analysis. Wavelet studies each component of the signal on different resolutions and scale. Wavelets capability to analyze the signal which contains sharp spikes and discontinuities

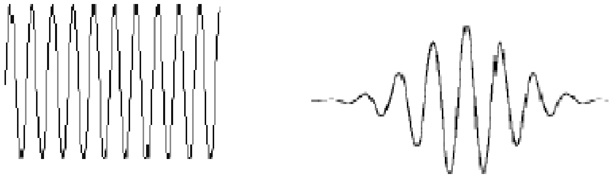


Fig:1 lossy and lossless wavelet transform

Early implementation of Wavelet transform based on the filters 'convolution algorithms' this approach requires a huge amount of computational resources. A relatively recent approach uses the Lifting scheme for the implementation of discrete wavelet transform (DWT). This method still constitutes an active area of research in mathematics and signal processing.

3. EXISTING ARCHITECTURE

The lifting scheme architecture is a technique for both designing wavelets and performing the discrete wavelet transform. Actually it is worthwhile to merge these steps and design the wavelet filters while performing the wavelet transform. This is then called the second generation wavelet transform. The technique was introduced by wim Sweldens.

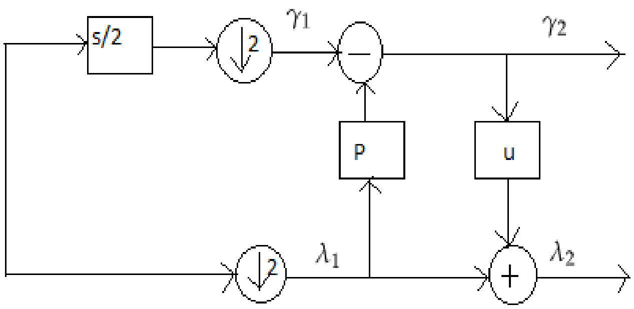


Fig: 2 Lifting scheme consists of two steps

Factorizes orthogonal and biorthogonal wavelet transforms into elementary spatial operators called lifting. It has two main applications. The first one is an acceleration of the fast wavelet transform algorithm. The filter bank convolution and sub sampling operations are factorized into elementary filtering on even and odd samples, which reduces the number of operations by nearly 2. Border treatments are also simplified. This is also called a para unitary filter bank implementation. The second application is the design of wavelets adapted to multidimensional bounded domains and surfaces, which are not possible with a Fourier transform approach. The discrete wavelet transform applies several filters separately to the same signal. In contrast to that, for the lifting scheme the signal is divided like a zipper. Then a series of convolution-accumulate operations across the divided signals is applied.

The simplest version of a forward wavelet transform expressed in the Lifting Scheme is shown in Figure P means predict step, which will considered in isolation. The predict step calculates the wavelet function in the wavelet transform. This is a high pass filter. The update step calculates the scaling function, which results in a smoother version of the data. As mentioned above the lifting scheme is an alternative technique for performing the DWT using bi-orthogonal wavelets. In order to perform the DWT using the lifting scheme the corresponding lifting and scaling steps must be derived from the bi-orthogonal wavelets. The analysis filters of the particular wavelet are first written in poly phase matrix form shown below.

$$\begin{bmatrix} h\,ev(z) & \cdots & g\,ev(z) \\ \vdots & \ddots & \vdots \\ h\,odd(z) & \cdots & g\,odd(z) \end{bmatrix} = p(z)$$

$$\begin{bmatrix} h\,ev(z) & g\,ev(z) \\ h\,odd(z) & g\,odd(z) \end{bmatrix} \begin{bmatrix} X\,ev(z) \\ X\,odd(z) \end{bmatrix}$$

$$= \begin{bmatrix} X\,1, L(z) \\ X\;\;1, H(z) \end{bmatrix}$$

$$Det[p(z)]=z^{-m}$$

The poly phase matrix is a 2 x 2 matrix containing the analysis low-pass and high-pass filters each split up into their even and odd polynomial coefficients and normalized. From here the matrix is factored into a series of 2 x 2 upper and lower triangular matrices each with diagonal entries equal to 1. The upper triangular matrices contain the coefficients for the predict steps and the lower triangular matrices contain the coefficients for the update steps. A matrix consisting of all 0's with the exception of the diagonal values may be extracted to derive the scaling step coefficients. The poly phase matrix is factored into the form shown in the equation below, a is the coefficient for the predict step and p is the coefficient for the update step.

$$P(z)= \begin{bmatrix} 1 & a(1-1/z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ b(1+z) & 1 \end{bmatrix}$$

$$P(z)= \begin{bmatrix} 1 & a(1-1/z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ b(1+z) & 1 \end{bmatrix} \text{x}$$

$$\begin{bmatrix} 1 & c(1-1/z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ d(1+z) & 1 \end{bmatrix} \text{x}$$

$$\begin{bmatrix} k1 & 0 \\ 0 & k2 \end{bmatrix}$$

An example of a more complicated extraction having multiple predict and update steps as well as scaling steps is shown below; a is the coefficient for the first predict step, p is the coefficient for the first update step, A, is the coefficient for the second predict step, 5 is the coefficient for the second update step, k1 is the odd sample scaling coefficient, and k 2 is the even sample scaling coefficient. according to matrix theory, any matrix having polynomial entries and a determinant of 1 can be factored as described above. Therefore every FTR wavelet or filter bank can be decomposed into a series of lifting and scaling steps. Daubechies and Sweldens discuss lifting step extraction in further detail.

CDF 9/7 Filter: A total of four lift steps are required, two predict and two update steps, to perform the proposed modified structure of CDF 9/7 DWT by adapting new scaling values of scaling coefficients of k1 and k2. The coefficients for predict and update steps are enlisted in table I.The predict and update equations for the CDF 9/7 filter are shown below.

Predict 1 : $odd_{new}=odd_{old}+[a(even_{left}+even_{right})]$

Update1: $even_{new}=even_{old}+[b(odd_{left}+odd_{right})]$

Predict2: $odd_{new}=odd_{old}+[c\,(even_{left}+even_{right})]$

Scale odd: $odd_{new} = [k1 * odd_{old}]$

Scale even: $odd_{new} = [k2 * odd_{old}]$

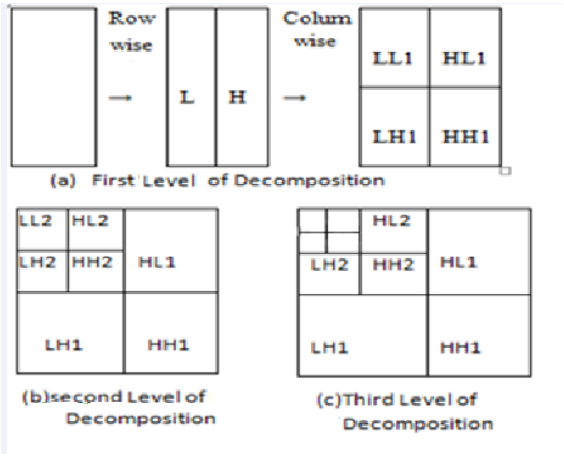
The floor function is used for all the predict, update and scale equations to provide an integer-to integer transform.The CDF 9/7 DWT consists of four lifting steps and two scaling steps. The first lifting step (predict step 1) is applied to the original row of samples and the results then safely overwrite the odd samples in the original signal for use in the next lifting step.

The discrete wavelet transform (DWT) is being increasingly used for image coding. This is due to the fact that DWT supports features like progressive image transmission (by quality, by resolution), ease of compressed image manipulation, region of interest coding, etc. DWT has traditionally been implemented by convolution. Such an implementation demands both a large number of computations and a large storage features that are not desirable for either high-speed or low-power applications. Recently, a lifting-based scheme that often requires far fewer computations has been proposed for the DWT. The main feature of the lifting based DWT scheme is to break up the high pass and low pass filters into a sequence of upper and lower triangular matrices and convert the filter implementation into banded matrix multiplications. Such a scheme has several advantages, including "in-place" computation of the DWT, integer-to-integer wavelet transform (IWT), symmetric forward and inverse transform, etc. Therefore, it comes as no surprise that lifting has been chosen in the upcoming.

There are two row processors to compute along the rows and two column processors to compute along the columns. While this arrangement is suitable or filters that require two banded-matrix multiplications filters that require four banded-matrix multiplications require all four processors to compute along the rows or along the columns. The outputs generated by the row and column processors (that are used for further computations) are stored in Memory module .

The memory modules are divided into multiple banks to accommodate high computational bandwidth requirements. The proposed architecture is an extension of the architecture for the forward transform that was presented. A number of architectures have been proposed for calculation of the convolution-based DWT. The architectures are mostly folded and can be broadly classified into serial architectures (where the inputs are supplied to the filters in a serial manner) and parallel architectures (where the inputs are supplied to the filters in a parallel manner). (a),(b)

Recently, a methodology for implementing lifting-based DWT that reduces the Memory requirements and communication between the processors, when the image is broken up into blocks.



(a) First Level of Decomposition

(b)second Level of Decomposition

(c)Third Level of Decomposition

Fig:3 Decomposition of DWT

For a system that consists of the lifting-based DWT transform followed by an embedded zero-tree algorithm, a new interleaving scheme that reduces the number of memory accesses has been proposed.

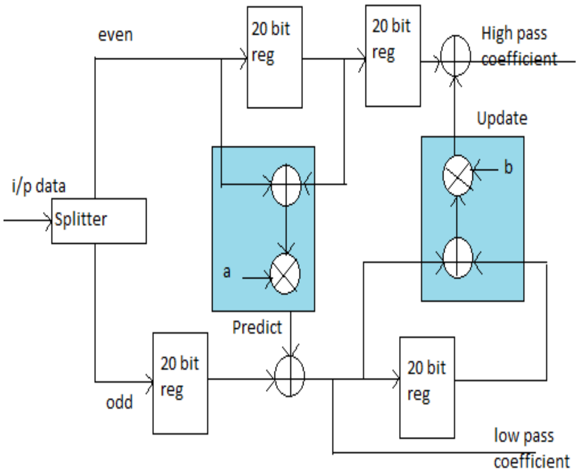


Fig:4 lifting scheme architecture

Finally, a lifting-based DWT architecture capable of performing filters with one lifting step, i.e., one predict and one update step. The outputs are generated in an interleaved fashion. The basic principle of the lifting scheme is to factorize the poly phase matrix of a wavelet filter into a sequence of alternating upper and lower triangular matrices and a diagonal matrix. This leads to the wavelet implementation by means of banded-matrix multiplications.

4. PROPOSED SCHEME

The proposed architecture for 3-D DWT consisting of two parallel spatial processors (2-D DWT) and four temporal processors (1-D DWT), is depicted in Fig. 1. After applying 2-D DWT on two consecutive frames, each spatial processor (SP) produces 4 sub-bands, viz. LL, HL, LH and HH which are fed to the inputs of four temporal processors (TPs) to perform the temporal transform. Output of these TPs consist of a low frequency frame (L-frame) and a high frequency frame (H-frame).

4.1 Architecture for Spatial Processor

In this section, we propose a new high-speed memory efficient lifting based 2-D DWT architecture denoted by spatial processor (SP).It consists of row and column processors. By introducing pipelining and sharing the computational load within the pipeline stages, the flipping based DWT equations proposed by Huang et al. in [12] are modified to eqns.(1)- (5). The proposed row processor (1-D DWT) and column processor (1-D DWT) have utilized the same equations ((1)- (5)) for implementation.

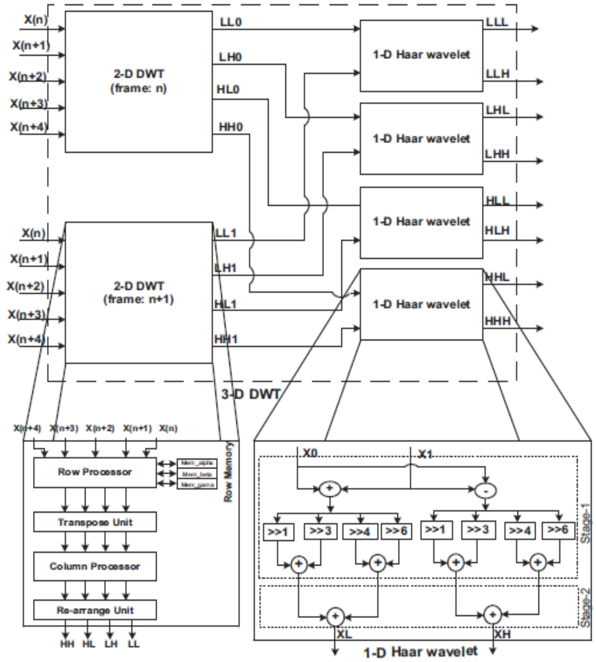


Fig:5 Proposed discrete wave let transform

$stage1$

$$X^{'}[2n-1] = 0.0078125 * X[2n-1]$$
$$X^{''}[2n-1] = 0.625 * X[2n-1]$$

$stage2$

$$a^{'} * X[2n-1] = X^{'}[2n-1] + X^{''}[2n-1]$$
$$H1[n] = a^{'} * X[2n-1] + X[2n] + X[2n-2]$$
$$A = b^{'} * X[2n] = 12 * X[2n]$$

$stage3$

$$B = H1^{'}[n] = 21 * H1[n]$$
$$C = H1^{''}[n] = 0.375 * H1[n]$$
$$L1[n] = A + H1[n] + H1[n-1]$$

$stage4$

$$D = d^{'} * L1[n] = L1^{'}[n] = 2.565 * L1[n]$$
$$H2[n] = L1[n] + L1[n-1] + B + C$$
$$H[n] = 0.0625 * H2[n]$$

$stage5$

$$L2[n] = D + H2[n] + H2[n-1]$$
$$L[n] = 0.03125 * L2[n]$$

The proposed architecture utilizes the strip based scanning [13] to enable the trade-off between external memory and internal memory. Pipelined lifting based DWT process has been performed by the processing unit (PU) in the proposed architecture. The proposed PU architecture reduces the CPD to 2Ta (two adder delay). From the internal architecture of the proposed PU. The number of inputs to the spatial processor is equal to 2P+1, which is also equal to the width of the strip, where P is the number of parallel processing units (PUs) in the row processor as well as column processor. The proposed architecture has been designed with two parallel processing units (P = 2). The same structure can be extended to P = 4, 8, 16 or 32 depending on external bandwidth. Whenever the row processor produces the intermediate results, the column processor starts to work on those intermediate results. Row processor takes 5 clocks to produce the temporary results, upon which the column processor takes 5 more clocks to to give the 2-D DWT output. Finally, temporal processor takes 2 more clock after 2-D DWT results are available to produce 3-D DWT output. Overall, the proposed 2-D DWT and 3-D DWT architectures have constant latency of 10 and 12 clock cycles respectively, regardless of the image size N and the number of parallel PUs (P).

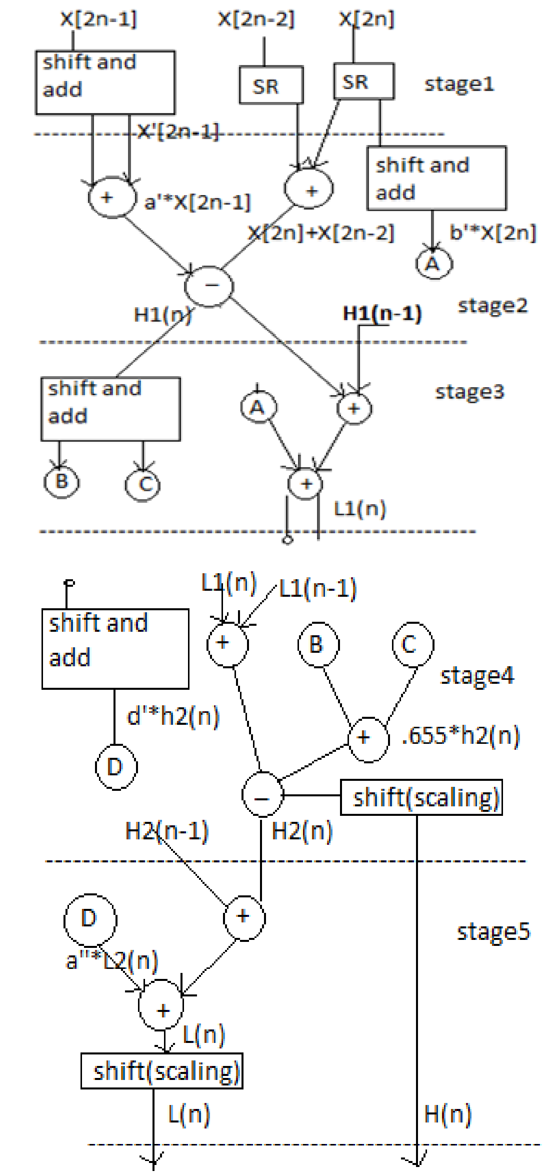


Fig: 6 data flow of DWT architecture

A detailed description of the row processor and the column processor is given in the following sub-sections.

4.1.1 Row Processor (RP): Let X be the image of size N×N, which is extended by one column using symmetric extension. Now the image size becomes N×(N+1). One may refer [12] for the structure of strip based scanning method. The proposed architecture initiates the DWT process row wise through the row processor (RP) and then process the column DWT by the column processor (CP). Fig. 2shows the generalized structure for a row processor with two (P=2) PUs. Each PU consists of five pipeline stages and each pipeline stage is processed by one processing element (PE) as depicted in Fig.4. The maximum CPD provided by these PEs is 2Ta. The outputs H1[n+1], L1[n+1], and H2[n+1] corresponding to PE alpha and PE beta of the last PU and PE gama of last PU is saved in the memories Memory alpha, Memory beta and Memory gama respectively. For an N×N image, the size of each memory is N×1 words and total row memory to store these outputs is 3N. Output of each PU is fed to the transposing unit, which has P number of transpose registers (one for each PU).Fig.3 shows the structure of the transpose register, which gives the two H and two L data alternatively to the column processor.

4.1.2 Column Processor (CP): The structure of the Column Processor (CP) is shown in Fig. 4. To match the RP throughput, CP is also designed with two PUs in our architecture. Each transpose register produces a pair of H and L in an alternative order, which are fed to the inputs of one PU of the CP. At the output of the CP, four sub bands are generated in an interleaved pattern, i.e., (HL,HH), (LL,LH), (HL,HH), (LL,LH), and so on. Outputs of the CP are fed to the re-arrange unit. Fig. 3(b) shows the architecture for the re-arrange unit, and it provides the outputs in sub-band order i.e LL, LH, HL and HH simultaneously.

4.2. Architecture for Temporal Processor

According to Sweldens all lifting based Haar wavelet transform depends on intensity values of two adjacent pixels. As soon as the spatial processors provide the 2-D DWT results, the temporal processors start processing on the spatial processor outputs and produce the 3-D DWT results. Fig. 1 shows that there is no requirement of temporal buffer, as the sub-band coefficients of two spatial processors are directly connected to the four temporal processors. Temporal processors apply 1-D Haar wavelet on sub-band coefficients, and provide the low frequency sub-band and high frequency sub-band as output.

5.RESULTS AND PERFORMANCE

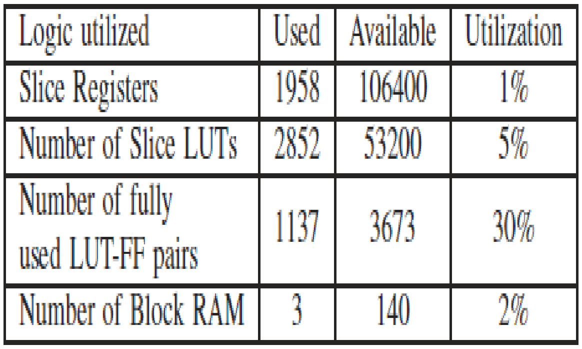
The proposed 3-D DWT architecture has been described in Verilog HDL. A uniform word length of 15 bits has been maintained throughout the design. Simulation results have been verified by using Xilinx ISE simulator. We have simulated the Matlab model which is similar to the proposed DWT hardware architecture and verified the DWT coefficients. RTL simulation results have been found to exactly match the Matlab simulation results. The Verilog RTL code is synthesized using Xilinx ISE 14.2 tool and mapped to a Xilinx programmable device (FPGA) with speed grade of -3. Table I shows the device utilization summary of the proposed architecture and it operates with a maximum frequency of 265 MHz. The proposed architecture has also been synthesized using synopsys design compiler with 90-nm technology CMOS standard cell library.

1. DWT Synthesis Result

This device utilization includes the following.

Logic Utilization
Logic Distribution
Total Gate count for the Design

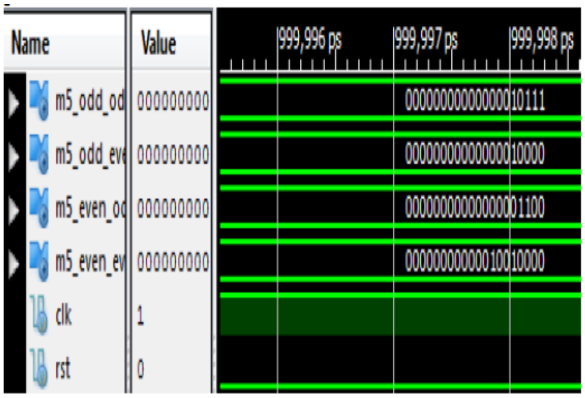
2. Device utilization summary:



Logic utilized	Used	Available	Utilization
Slice Registers	1958	106400	1%
Number of Slice LUTs	2852	53200	5%
Number of fully used LUT-FF pairs	1137	3673	30%
Number of Block RAM	3	140	2%

Table-1: Device utilization

Simulation results :



Name	Value
m5_odd_od	000000000
m5_odd_eve	000000000
m5_even_od	000000000
m5_even_ev	000000000
clk	1
rst	0

VI. CONCLUSIONS

In this paper, we have proposed high-speed and memory efficient architecture for lifting based DWT. The proposed architecture has not only been implemented on FPGA target of zinc family, but has also been synthesized on Synopsys' design vision for ASIC implementation. An efficient design of 2-D spatial processor and 1-D temporal processor reduces the internal memory, latency, critical path delay and complexity of a control unit, and increases the throughput. When compared with several existing architectures, the proposed scheme shows higher performance at the cost of a slight increase in area.

References:

1. Akansu, Ali N.; Haddad, Richard A. (1992), Multiresolution signal decomposition:

2 A. G. Ramakrishnan and S. Saha, "ECG coding by wavelet-based linear prediction,"transforms, sub bands, and wavelets, Boston, MA: Academic Press, ISBN978-0-12-047141-6,IEEETrans. Biomed. Eng., Vol. 44, No. 12, pp. 1253-1261, 1977.

3. N.Malmurugan, A. Shanmugam, S. Jayaraman and V. V. Dinesh Chander. "A New and Novel Image Compression Algorithm Using Wavelet Footprints"

4 Ho Tatt Wei and Jeoti, V. "A wavelet footprints-based compression scheme for ECG signals". Ho Tatt Wei; Jeoti,V. (2004)."A wavelet footprints-based compression scheme for ECG signals".2004 IEEE Region 10 Conference TENCON 2004. A. 283. doi:10.1109/TENCON.2004.1414412. ISBN 0 -7803-8560-8.

5. J. E. Fowler and J. T. Rucker, "3-D wavelet-based compression of hyper spectral imagery," in Hyper spectral Data Exploitation: Theory and Applications, C.-I. Chang, Ed. Hoboken, NJ: Wiley, 2007, ch. 14,pp. 379–407.

6. A. Skodras, C. Hristopoulos, and T. Ibrahim, "The JPEG 2000 still image compression standard," IEEE Signal Process. Mag., vol. 18, no. 5, pp. 36–58, Sep. 2001.

7. W. Badawy, G. Zhang, M. Talley, M. Weeks, and M.Bayoumi, "Low power architecture of running 3-D wavelet transform for medical imaging application," in Proc. IEEE Workshop Signal Process. Syst., Taiwan,1999, . 65–74.

8. B. Girod and S. Han, "Optimum update for motion-compensated lifting," IEEE Signal Process. Lett., vol. 12, no. 2, pp. 150–153, Feb. 2005.

9.C. Pilrisot, M. Antonini, and M. Barlaud, "3-D scan based wavelet transform and quality control for video coding," Eur. Assoc. Signal Process. J. Appl. Signal Process., vol. 2003, pp. 56–65, Jan. 2003.

[10] B. K. Mohanty and P. K. Meher, "Memory-Efficient Architecture for 3-D DWT Using Overlapped Grouping of Frames," IEEE Transactions on Signal Processing, vol. 59, no. 11, pp.5605-5616, Nov. 2011.

[11] A. Darji, S. Shukla, S. N. Merchant and A. N. Chandorkar, "Hardware Efficient VLSI Architecture for 3-D Discrete Wavelet Transform," Proc. of 27th Int. Conf. on VLSI Design and 13th Int. Conf. on Embedded Systems pp. 348-352, 5-9 Jan. 2014.

[12] C.T. Huang, P.C. Tseng, and L.G. Chen, "Flipping structure: an efficient VLSI architecture for lifting-based

discrete wavelet transform," IEEE Trans. Signal Process., vol. 52, no. 4, pp. 1080-1089, Apr. 2004.

[13] C. Y. Xiong, J. W. Tian, and J. Liu, "A Note on Flipping Structure: An Efficient VLSI Architecture for Lifting-Based Discrete Wavelet Transform," IEEE Transactions on Signal Processing, vol. 54, no. 5, pp. 1910-1916, May 2006

[14] Y. Hu and C. C. Jong, "A Memory-Efficient Scalable Architecture for Lifting-Based Discrete Wavelet Transform," IEEE Transactions on Circuits and Systems-II: Express Briefs, vol. 60, no. 8, pp. 502-506, Aug. 2013.

[15] W. Sweldens, "The Lifting Scheme: a Construction of Second Generation of Wavelets," SIAM Journal on Mathematical Analysis, vol.29 no.2, pp. 511-546, 1998.

[16] C.T. Huang, P.C. Tseng, and L.-G. Chen, "Analysis and VLSI architecture for 1-D and 2-D discrete wavelet transform," IEEE Trans. Signal Process., vol. 53, no. 4, pp. 1575-1586, Apr. 2005.