

## Power Efficient Vlsi Architecture For High Throughput And Low Area Implementation Of Lifting 2-D Dwt

J.Muni Dhananjaya Babu<sup>1</sup>, Mrs.B.Sudha<sup>2</sup>

<sup>1</sup>(Department of Electronics and Communication Engineering, SRM University, India)

<sup>2</sup>(Department of Electronics and Communication Engineering, SRM University, India)

**ABSTRACT:** In this paper a new VLSI architecture was implemented for computation of lifting two-dimensional (2-D) discrete wavelet transform (DWT). This structure was data transposition free and it was implemented by using linear systolic array which is derived from the dependence graph (DG) and a 2-D systolic array from a segmented DG for implementation of 1-D DWT. By using these two systolic arrays a new structure for was implemented for calculation of lifting 2-D DWT. The proposed structure does not require any control signal. The synthesis results shows that the proposed structure required less gate count and power consumption and high throughput compared to existing structure. This structure was regular and it can be easily configured to different block sizes.

**Keywords -** Discrete wavelet transform (DWT), Lifting, Systolic array.

### I. INTRODUCTION

The discrete wavelet transform (DWT) has many applications in signal analysis, computer vision, object recognition, image compression and video compression. Two-Dimensional (2-D) discrete wavelet transform (DWT) is extensively used in image and video compression. Image compression is the art of science of reducing the amount of data required to represent an image. Image compression is important in order to reduce the storage need. VLSI implementation of 2-D DWT is subjected to a set of constrains for example the silicon area and power consumption as well as minimum processing speed for real time computation. There are many architectures were proposed for efficient implementation of 2-D DWT for real time applications. Those structures are classified in to 2 types. There are (1) convolution based and (2) lifting based. The convolution based designs required more arithmetic and memory resources than the lifting based designs.

The hardware complexity of the 2-D structures based on arithmetic component and memory component. The arithmetic component includes multipliers and adders on the other hand memory component includes transposition memory and temporal memory. Transposition memory is used for storing of inputs and intermediate coefficients, while the temporal memory is used for storing of partial results of filter output. The sizes of the transposition and temporal memory depends on the width of the input image, while the hardware complexity of arithmetic component depends on the wavelet filter size and computation scheme used either lifting/convolution. The sizes of the image is high compared to size of the wavelet filter so complexity of memory component is the main issue for the complexity of 2-D DWT structures. So in this paper proposed a transposition memory free structure.

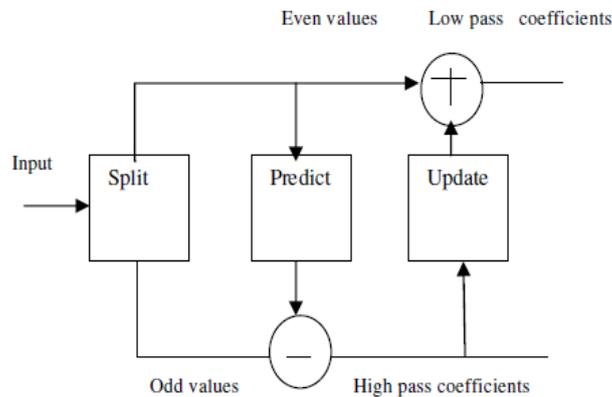
### II. LIFTING SCHEME

The lifting scheme was first implemented by Swelden's . The lifting scheme[1] entirely depends on the spatial domain. This scheme has so many advantages compared to other schemes such has lower area, power consumption and computational complexity. The basic lifting scheme consisting of three basic steps 1. Splits 2. Predict 3. Update, which is shown in below fig.1.

1. Splits: In this step the signal is divided in to even and odd points. For the given input samples  $x(n)$  split into even  $x(2n)$  and odd coefficients  $x(2n+1)$ .

2. Predict: In this step the even samples are multiplied by the predict factor and added to the odd samples to generate the detailed coefficients which results in high pass filtering. The equation for predict step is given as (1).

$$HP[2n + 1] - X[2n - 1] - \frac{[X[2n] + X[2n + 1]]}{2} \dots(1)$$



**Fig.1. Basic Lifting Scheme.**

3. Update step: In this step detailed coefficients are multiplied by the update factors and then added to the even samples to get coarser coefficients which gives low pass filtered output.

$$LP[2n] = X[2n] + \frac{HP[2n-1] + HP[2n+1] + 2}{4} \dots\dots(2)$$

### III. SYSTOLIC ARRAY FOR LIFTING 1-D DWT

In systolic array the processors are arranged in such a way that, the data flows synchronously across the array between neighbors, usually the different data flowing in different directions. Each processor at each step takes in data from one or more neighbors (e.g. North and West), processes it and, in the next step, outputs results in the opposite direction (South and East). In systolic arrays the processors are connected by short wires. Lifting computation of 1-D DWT is expressed as

$$\begin{aligned} s1(n) &= x(2n-1) + \alpha((x(2n) + x(2n-2))) \\ s2(n) &= x(2n-2) + \beta((s1(n) + s1(n-1))) \\ uH(n) &= s1(n-1) + \gamma((s2(n) + s2(n-1))) \\ uL(n) &= s2(n-1) + \delta((uH(n) + uH(n-1))) \end{aligned} \quad (3)$$

Where  $\alpha, \beta, \gamma, \delta$  are lifting constants. Those values are  $\alpha \approx -1.586134342$ ,  $\beta \approx -0.052980118$ ,  $\gamma \approx 0.8829110762$ ,  $\delta \approx 0.4435068522$ .  $x(n)$  is the input sample values,  $uL(n)$  and  $uH(n)$  represent the low-pass and high-pass intermediate outputs scaled by constants  $K$  and  $\frac{1}{K}$ .

Data dependence graph (DG) of N-point lifting DWT is shown in below fig.2. It consists of  $N/2$  identical sections, and each section consists of four identical nodes.

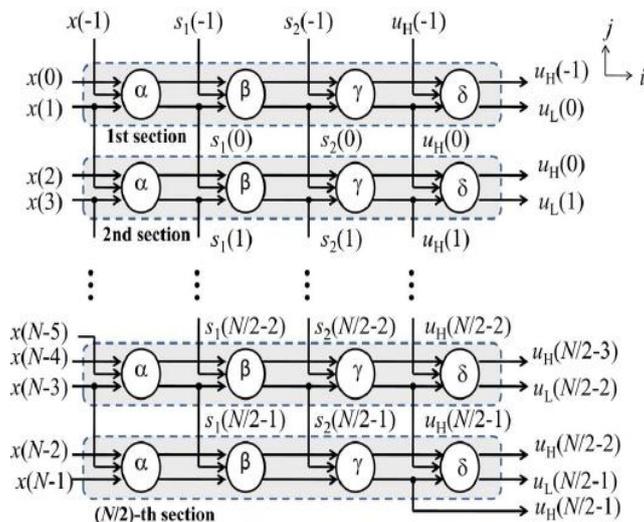


Fig.2. DG of N-point lifting DWT.

### 3.1 One –Dimensional systolic array

By projecting the nodes of above dependence graph shown in figure 2, and using a systolic schedule we can get a 1-D systolic array shown in below fig.3, which consists of four processing elements (PE) and process two samples in every cycle.

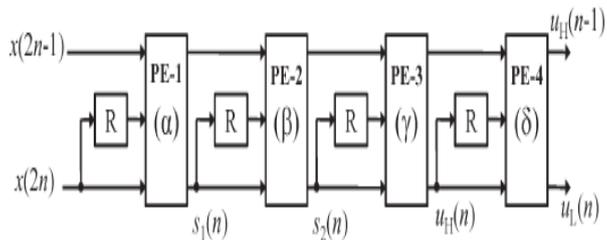


Fig.3 One – Dimensional systolic array

The function of each processing element is shown in fig.4. It consists of one multiplier and two adders. The cycle period is  $T=T_m+2T_a$ , where  $T_m$  is the time required to perform one multiplication and  $T_a$  is the time required to perform one addition in a processing element. The 1-D systolic array takes  $N/2$  cycles for calculation of N-point 1-D dwt.

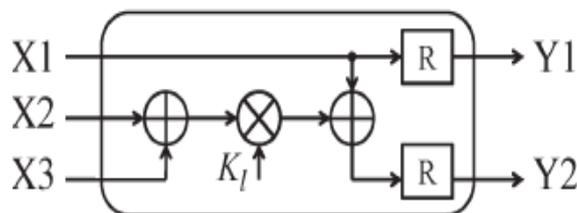
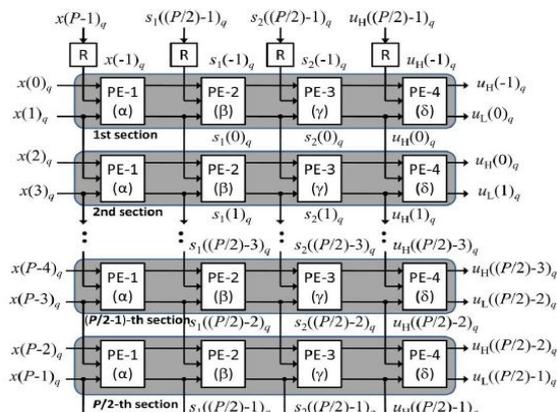


Fig.4.Processing element

### 3.2.Two dimensional systolic array

The 2-D systolic array shown in below fig.5, which consists of  $2P$  processing elements arranged in  $P/2$  rows and four columns where  $P$  is the block size.

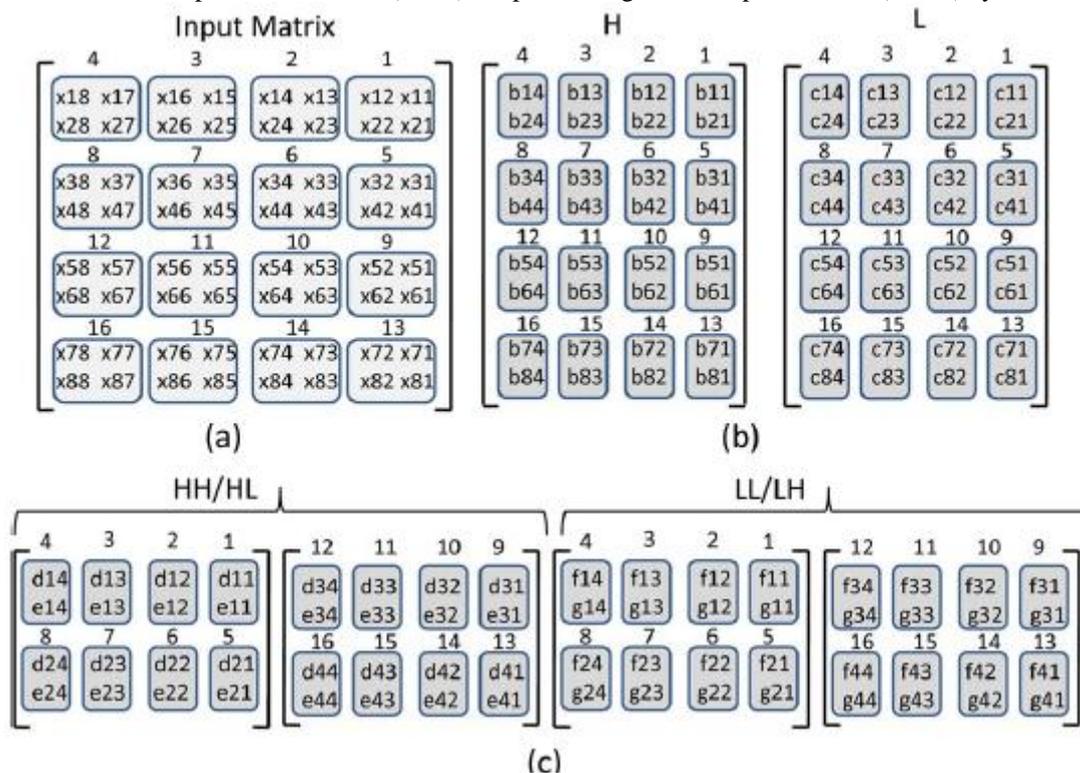


**Fig.5 Two –Dimensional systolic array for N-point 1-D lifting DWT**

The 2- D array consists of four registers to store the intermediate values. In each cycle, it takes a block of P input samples and calculate P/2 low-pass and P/2 high-pass DWT components.

#### IV. PROPOSED DATA ACCESS SCHEME

In the proposed data access scheme by using embedded down sampling intermediate coefficients will generate. The input blocks of size P are prepared from P/2 rows and two adjacent columns. Let us take block size (P=4) then for 8×8 image the data access scheme will be illustrated as shown in fig. 6. In this scheme the first four data blocks are prepared by using first two rows, next set of four blocks are prepared from third and fourth rows. Each data block contains samples corresponding to two consecutive rows and 2 columns. The data blocks are pass through the row processor serially, the data blocks of first P/2 rows pass through the first set of N/2 clock cycles, then the data blocks of next P/2 rows are pass through the second set of N/2 cycles. Data blocks of the entire input matrix of size (M×N) are pass through the row processor in (MN/P) cycles.



**Fig.6. (a) Proposed data access scheme for input matrix [8×8] and block size P=4. (b) Data flow of row processor/column processor (c) output data flow of the column processor.**

After processing each data block, the row processor produce intermediate matrices  $uL$  and  $uH$  of size  $(P/2)$ . The data block of matrices  $uL$  and  $uH$  belong to the same column and these data blocks can be processed by the column processor without transposition. Hence, data blocks of  $uL$  and  $uH$  are processed by the column processor in the same order as generated by the row processor.

### V. PROPOSED LIFTING 2-D DWT STRUCTURE

The proposed structure is shown in fig.7, which consists of one row processor and one column processor. The row processor consisting of  $P/2$  1-D systolic arrays shown in figure3. Where  $P$  is the block size. Data blocks are pass through the processor as per the format shown in fig. 6(a).each 1-D array of the row processor receives a pair of samples corresponding to a row. The row processor receives  $(N/2)$  data blocks corresponding to  $(P/2)$  rows in  $N/2$  cycles. 1-D array of the row processor calculates one low-pass and one high-pass intermediate coefficients in every cycle. A block of  $(P/2)$  components of every column of  $uL$  and  $uH$  is obtained from the row processor in every cycle. These intermediate data blocks are directly pass through the column processor without data transposition.

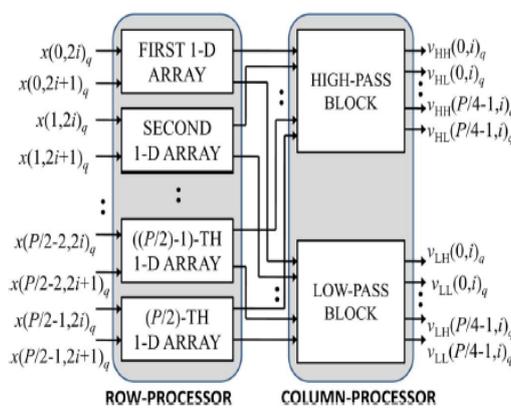


Fig.7. Proposed structure

The column processor composed of one high-pass block and one low-pass block. The high-pass block processes the high-pass intermediate matrix and generates sub band matrices  $vHL$  and  $vHH$ . The low-pass block processes the low-pass intermediate matrix and generates the sub band matrices  $vLH$  and  $vLL$ . The high-pass block and low-pass block consisting of 2-D systolic array shown in figure 5. The four registers of the 2-D systolic array are replaced by four shift registers of  $N/2$  words each to have a low-pass and high-pass block. Shift registers are used to storing the intermediate coefficients and partial results. The low-pass and high pass outputs of row and column processor are scaled according to the lifting scheme for the 9/7 filter. The low-pass block of column processor computes  $P/4$  components of each sub bands ( $vLL$  and  $vLH$ ) in every cycle during the same period the high-pass block calculates  $P/4$  rows of two sub bands in  $N/2$  cycles. The column processor generates four sub band matrices ( $vLL$ ,  $vLH$ ,  $vHL$ ,  $vHH$ ). In this proposed structure both row processor and column processor work con currently and computes 1-level lifting 2-D DWT of an image size  $(M \times N)$ .

### VI. SIMULATION AND SYNTHESIS RESULTS

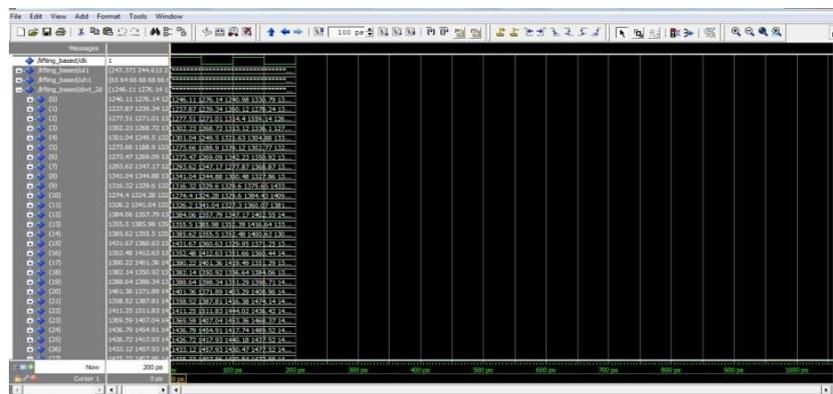
The proposed design as well as existing design [2] were simulated in VHDL and synthesis is performed in Xilinx ISE . For the proposed design in the place of multiplier booth multiplier was used while in the case of adder carry save adder was used in synthesis.

The input image as shown in fig.8 is given as input to MATLAB then MATLAB converts that image pixels values in to text file, that text file fed as input to the MODELSIM VHDL code for the proposed structure after compiling the code we can get image pixel values in the form of text file. By using MATLAB we can get compressed output image.



**Fig.8 Input image to MATLAB**

The simulation results of DWT of Model Sim of image in fig 8 is shown in fig 9.for proposed block size(P=4).



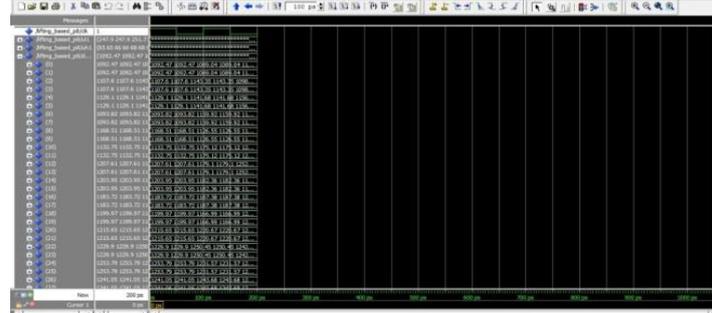
**Fig.9 Model sim output for proposed block size (P)=4**

After interfacing the model sim output to matlab the compressed output image for proposed block size (P)=4 is as shown in below fig.10.



**Fig.10 Compressed output image for proposed block size(P)=4.**

The simulation results of DWT of Model Sim of image in fig.8 is shown in fig.11,for proposed block size(P)=8.



**Fig.11 Model sim output for proposed block size(P)=8.**

After interfacing the modelsim output to matlab the output image for proposed block size (P)=4 is as shown in below fig.12.



**Fig.12 Compressed output image for proposed block size (P)= 8**

### 6.1 Synthesis Results

The synthesis was performed on Xilinx ISE for both existing [2] and proposed structures the following table illustrates the comparison between existing and proposed structures.

Structure	Gate count	Power(mw)	Delay(ns)
Existing[2]	9298	74	8.12
Proposed block size4(P=4)	6497	53	7.12
Proposed block size(P=8)	9968	64	6.24

**Table1.Comparison between existing and proposed lifting 2-D dwt structures**

## VII. CONCLUSION

The proposed VLSI structure for computing lifting 2-D DWT (Discrete wavelet transform) with out data transposition was implemented . The proposed structure was simulated and then synthesized. The synthesis results were compared to the existing DWT structure and it shows that the proposed structure consumes less area, power consumption and delay. The proposed structure for 2-D lifting DWT was regular, modular and it can be easily configure for different block sizes.

## REFERENCES

- [1]. G. Shi, W. Liu, and L. Zhang, "An efficient folded architecture for lifting based discrete wavelet transform," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 56, no. 4, pp. 290–294, Apr.2009.
- [2]. 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* vol.9 No.3, march 2009.
- [3]. W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," *Appl. Comput. Harmon. Anal.*, vol. 3, no. 2, pp. 186– 200, Apr. 1996.
- [4]. P.-C. Wu and L.-G. Chen, "An efficient architecture for two-dimensional discrete wavelet transform," *IEEE Trans. Circuits Syst. Video Technol.*,vol. 11, no. 4, pp. 536–545, Apr. 2001.
- [5]. 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.
- [6]. C.-Y. Xiong, J.-W. Tian, and J. Liu, "Efficient architecture for two dimensional discrete wavelet transform using lifting scheme," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 607–614, Mar. 2007.
- [7]. 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.
- [8]. X. Tian, L. Wu, Y.-H. Tan, and J.-W. Tian, "Efficient multi-input/multioutputVLSI architecture for two-dimensional lifting-based discrete wavelet transform," *IEEE Trans. Comput.*, vol. 60, no. 8, pp. 1207–1211, Aug. 2011.