iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 4 **Issue:** III **Month of publication:** March 2016

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Efficient Multiplier Design for DWT through Lifting Scheme

Ms. Dhrisya[1], Mr. V Lakshmipathi[2]
M.E VLSI Design[1], Professor & Head, ECE Department[2]
JCT College of Engineering & Technology, Pichanur, Coimbatore, Tamil Nadu

Abstract- For having loss less data transmission in the communication world, with the aid of lifting based algorithm we go for wavelet transform technique. A single signal (speech, image or audio) is converted into high frequency and low frequency components. As the high frequency signal contains noise, it is not further processed. The low frequency signal is subdivided by using sub band coding. The Discrete Wavelet Transform (DWT) plays a major role in the fields of signal analysis, computer vision. DWT needs extra memory for storing the intermediate computational results. For real time image compression, DWT has to process massive amounts of data at high speeds. Hardware implementation of DWT has many practical obstacles. The Filter bank analysis of wavelet transforms is in the frequency domain and not in the time domain and the filter coefficients are not integer numbers so they are not appropriate for hardware implementation. So a multiplier has to be designed for efficient hardware implementation. In this paper a Wallace tree multiplier is designed for DWT through lifting scheme. The lifting based DWT does not depend on the Fourier transform, so the calculation of wavelets gets speedup. Simulation of this project is done using Xilinx ISE simulator. Also Power consumption and delay time is calculated, with the aid of lifting based.
Keywords: Discrete wavelets transform (DWT), Image compression, Wallace tree multiplier, Lifting scheme, Simulation

I. INTRODUCTION

The Discrete Wavelet Transform (DWT), which is based on sub-band coding, is found to yield a fast computation of Wavelet Transform. It is easy to implement and reduces the computation time and resources required. The Discrete Wavelet Transform (DWT) plays a major role in the fields of signal analysis, computer vision, object recognition, image compression and video compression standard. The advantage of DWT over other traditional transformations is that it performs multi resolution analysis of signals with localization both in time and frequency. The Discrete wavelet transform (DWT) is a multi-resolution analysis tool with excellent characteristics in the time and frequency domains. Through the DWT, signals can be decomposed into different sub bands with both time and frequency information. The coding efficiency and the quality of image restoration with the DWT are higher than those with the traditional discrete cosine transform. Moreover, it is easy to obtain a high compression ratio. As a result, the DWT is widely used in signal processing and image compression, such as MPEG-4, JPEG2000; and so on Traditional DWT architectures are based on convolutions

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, ease of compressed image manipulation, region of interest coding etc. A general fashion in which DWT decomposes the input image is shown below in Fig.1.

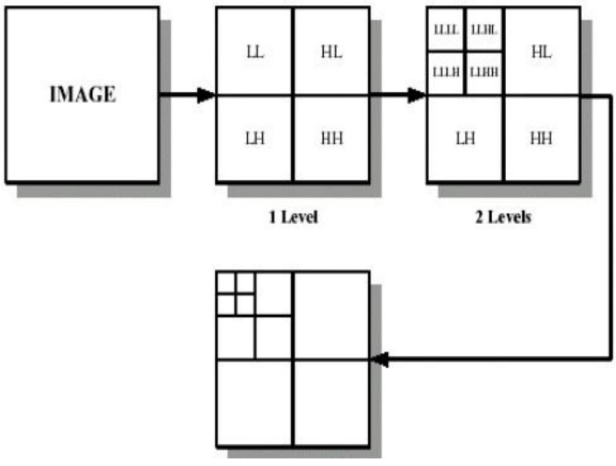


Fig.1.Three level decomposition of an image

Each decomposition level shown in Fig.2 comprises two stages: stage 1 performs horizontal filtering and stage 2 performs vertical filtering. In the first-level decomposition, the size of the input image is N* N, and the outputs are the three sub bands LH, HL, and HH, of size N/2*N/2. In the second-level decomposition, the input is the LL band and the outputs are the three sub bands LLLH, LLHL, and LLHH, of size N/4*N/4.The multi-level 2-D DWT can be extended in an analogous manner. The arithmetic computation of DWT can be expressed as basic filter convolution and down sampling. Two dimensional discrete wavelet transform (DWT) is defined as:

$$X_{LL}^{J}(n_1,n_2)= \sum_{i_1=0}^{k-1}\sum_{i_2=0}^{k-1} g(i_1).\, g(i_2).X_{LL}^{J-1}(2n_1 - i_1)\,(2n_2 - i_2 \quad (1)$$

$$X_{LH}^{J}(n_1,n_2) = \sum_{i_1=0}^{k-1}\sum_{i_2=0}^{k-1} g(i_1).\mathrm{h}(i_2).X_{LL}^{J-1}(2n_1 - i_1)(2n_2 - i_2) \quad (2)$$

$$X_{HL}^{J}(n_1,n_2) = \sum_{i_1=0}^{k-1}\sum_{i_2=0}^{k-1} h(i_1).\, g(i_2).X_{LL}^{J-1}(2n_1 - i_1)(2n_2 - i_2) \quad (3)$$

$$X_{HH}^{J}(n_1,n_2) = \sum_{i_1=0}^{k-1}\sum_{i_2=0}^{k-1} h(i_1).\, h(i_2).\, X_{LL}^{J-1}(2n_1 - i_1)\,(2n_2 - i_2) \quad (4)$$

Where ,X_{LL} (n1, n2) is the input image, J is 2-D DWT level is filter length, g (n) is impulse responses of the low-pass filter and h (n) impulse responses of the high-pass filter.

II. HARDWARE REQUIREMENTS OF LIFTING SCHEME

The three basic steps in Lifting based DWT are:

Split step: where the signal is split into even and odd points, because the maximum correlation between adjacent pixels can be utilized for the next predict step. For each pair of given input samples x (n) split into even x (2n) and odd coefficients x (2n+1)

Predict step: The even samples are multiplied by the predict factor and then the results are added to the odd samples to generate the detailed coefficients (dj).Detailed coefficients results in high pass filtering.

$$\mathrm{HP}\,[2n+1] = \mathrm{X}\,[2n-1] - \frac{\mathrm{X}[2n]+\mathrm{X}[2n+2]}{2} \quad (5)$$

Update step: The detailed coefficients computed by the predict step are multiplied by the update factors and then the results are added to the even samples to get the coarse coefficients (sj).The coarser coefficients gives low pass filtered output.

$$\mathrm{LP}\,[2n] = \mathrm{X}\,[2n] + \frac{\mathrm{HP}\,[2n-1]+\mathrm{HP}\,[2n+1]+2}{4} \quad (6)$$

Fig.2. shows the lifting scheme of the wavelet filter computing one dimension signal. The inverse transform could easily be found by exchanging the sign of the predict step and the update step and apply all operations in reverse order as shown in Fig.3. The implementation of lifting based inverse transform (IDWT) is simple and it involves order of operations in DWT to be reversed. Hence the same resources can be reused to define a general programmable architecture for forward and inverse DWT.

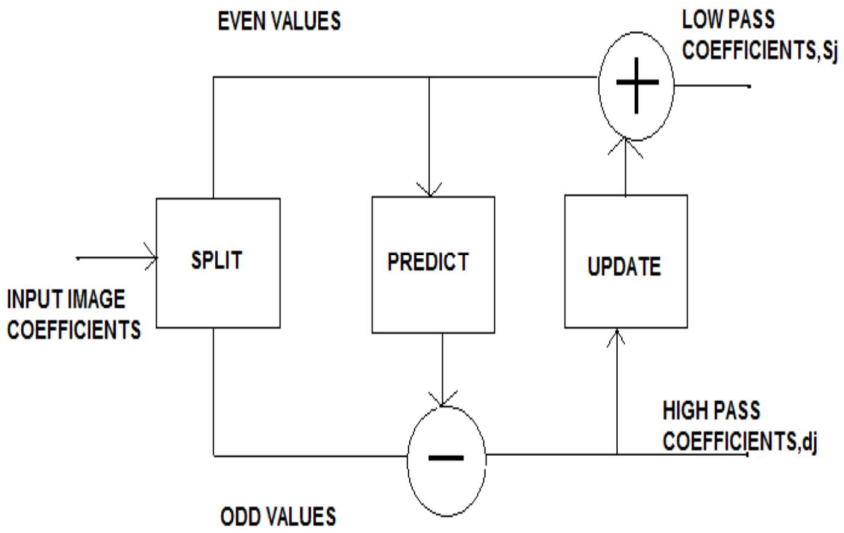


Fig.2. Block diagram of forward lifting scheme

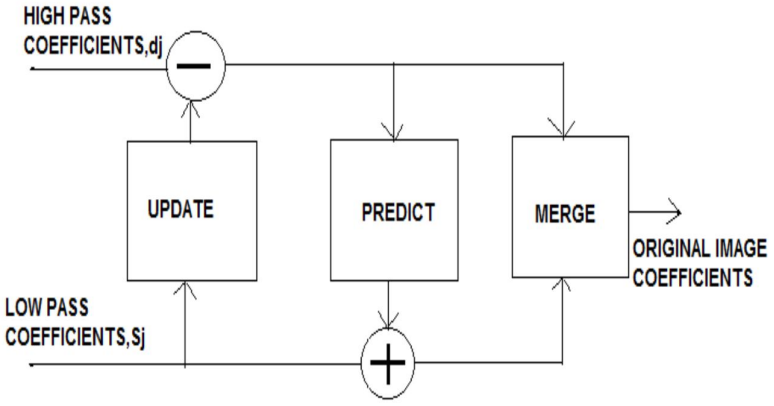


Fig.3. Block diagram of inverse lifting scheme

A. Basic Functional Units of Lifting Scheme

Different kinds of lifting-based DWT architectures can be constructed by combining the three basic lifting elements. Most of the applicable DWTs like (9, 7) and (5, 3) wavelets consist of processing units, as shown in Fig.5. This unit is called the processing element (PE). The processing nodes A, B and C are input samples which arrive successively. To implement the predict unit, A and C receive even samples while B receives odd samples. On the other hand, for the update unit, A and C are odd samples and B receives even samples. Now, the structure can be used to implement (5, 3) wavelets is shown in Fig.4. In this architecture each white circle represents a PE. The low pass filter has 5 taps and the high pass has 3 taps and hence it is (5, 3) wavelet layers are essential (basic) layers and are fixed for each wavelet type, while by changing the number of extended layers, the type of wavelet can be changed accordingly. The black circles represent needed stored data for computing outputs (s, d). R0, R1 and R2, are registers that get their values from new input samples and are called data memory. The input and output other three black circles which store the results of previous computations are known as temporary memory.

Table.1 Complexity comparison of convolution and lifting DWT

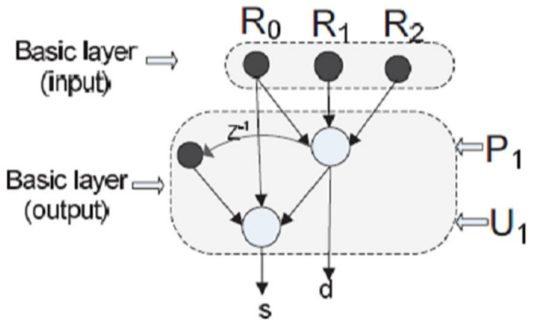


Fig.4.Lifting structure for (5, 3) wavelet

The number of data memory registers is constant and is equal to 3, while the number of temporary memory registers is ($2e + 1$), where $e$is the number of extended layers. This structure can be implemented by using combinatorial circuits so that, when the input samples are fed to the architecture, outputs are ready to be used after a delay time. Also, the implementation of the structure can be performed via a pipelined structure by adding some registers. The number of pipeline stages depends on the added registers. Increasing the pipeline stages increases the clock frequency, system latency and number of required registers. So, the sum of the data and temporary memories in the column-wise DWT unit determines the amount of needed internal memory. The pipeline registers do not affect the required internal memory. The pipeline registers do not affect the required internal memory. From the Table.1 it is clear that the lifting scheme will be more suitable for hardware implementation of DWT with limited on-chip memory, lower computational complexity, small area and low power.

B. Mathematics in Lifting and DWT

DWT of perfect reconstruction filters can be decomposed into a finite sequence of lifting steps. The decomposition corresponds to a factorization of the poly-phase matrix of the target wavelet filter into a sequence of alternating upper and lower triangular

matrices and a constant diagonal matrix,

$$h(z) = h_e(z^2) + z^{-1}h_o(z) \quad (7)$$
$$g(z) = g_e(z^2) + z^{-1}g_o(z) \quad (8)$$

$$P(z) = \begin{vmatrix} h_e(z) & g_e(z) \\ h_o(z) & g_o(z) \end{vmatrix} \quad (9)$$

P (z) is called the dual (synthesis) of$\tilde{p}$ (z) and for perfect reconstruction p (z) $(z^{-1})^T$ =I where I is the 2x2 identity matrix. Where h_e and h_o are even and odd taps of LPF, g_e and g_o are even and odd taps of HPF.

FILTER	MULTIPLICATION /SHIFTS		ADDITION	
	CONVOLUTION	LIFTING	CONVOLUTION	LIFTING
(5,3) LOSE LESS	4	2	6	4

C. Minimizing the Hardware Architectures-Parallel and Direct Mapped Structures

For lifting schemes that require only 2 lifting steps, such as the (5, 3) filter consists of two pipeline stages. The architecture can be sequentially pipelined by combining the previous output of predict stage to current output.

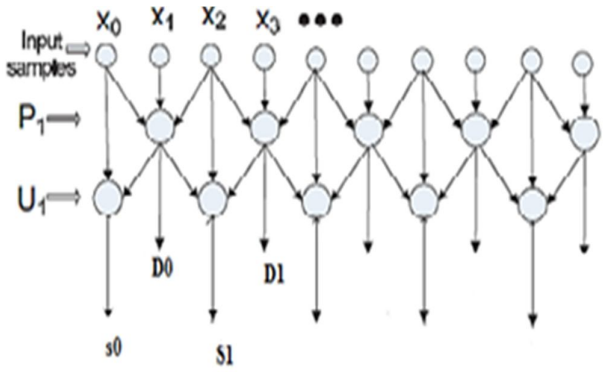


Fig.5. DWT architecture for (5, 3) wavelet based on parallel filter method

In this structure, U1 (0) represents the current output of the U1 unit and P1 (-1) represents the previous output of the P1 unit, and so on. The control signal **S**, which has four states, selects the inputs of the multiplexers sequentially. In the first state, two consecutive input samples arrive and the P1 function with **a** coefficient is performed on them. In the second state, the U1 function with **b** coefficient will be imposed on the result of the previous state (first state's output).

Table.2 Data flow for (5, 3) hardware architecture

S	IN1	IN2	OUT	F (factor)
0	I0	I1	P1	A
1	I0(-1)	P1	U1	B

D. Boundary Treatment of Signals

For a lifting-based the signals should first be extended periodically as shown in Fig.6. This periodic symmetric extension is used to ensure that for the filtering operations that take place at both boundaries of the signal, one signal sample exists and spatially corresponds to each coefficient of the filter mask. The number of additional samples required at the boundaries of the signal is therefore filter-length dependent. In this method the signal is extended so as it becomes periodic and symmetric. The mirror extension can be seen on the left and on the right of the original line (A-B-C-D-E). The values of I left and I right parameters are chosen in relation to the filter length. This table explains, for example, that if I0 is even, we have to extend the line with 1 coefficient to the left when (5, 3) transform is used, or 3 coefficients to the left with the (9, 7) transform. In VLSI implementation the embedded mirror symmetric is implemented into the data by changing the operation process at the beginning and end of the lifting operation using multiplexers.

Fig.6. Boundary treatments of signals

III. MULTIPLIER

Multipliers form a major part of DSP applications. The multipliers are used in 2D-DWT which is used for image compression and signal processing applications. Any multiplier design has 3 steps. They are

Partial product generation

Partial product reduction

Final addition.

The partial products are formed first either by using modified booth algorithm or by ANDing each bit of the multiplier with each bit of the multiplicand. The next step is reduction of these partial products to two rows. This is done by grouping the partial products into groups of three. These groups are reduced by using carry save adders. Carry save adders are mainly full adders and half adders. The third step is addition of the final two rows by using carry look ahead adders to yield the final product.

A. Wallace Tree Multiplier

A Wallace tree is an efficient hardware implementation of a digital circuit that multiplies two integers. The Wallace tree is used for reduction of partial products. The firststep is ANDing of each bit of the multiplier with each bit of the multiplicand to yield the partial product. The next step is reduction of partial products to two rows. This is done by grouping the partial products to rows of three. There are 3 rules in partial product reduction

Apply a full adder to each column which contains 3 bits

Apply a half adder to each column which contains 2 bits

A single bit column is passed on as such.

After the reduction we get two rows. These two rows are added using carry save adders.

1) Advantages: Each layer of the tree reduces the number of vectors by a factor of 3:2

Minimum propagation delay.

The benefit of the Wallace tree is that there are only O (log n) reduction layers, but adding partial products with regular adders would require O (log n) 2 times.

2) Disadvantages: Wallace trees do not provide any advantage over ripple adder trees in many FPGAs.

Due to the irregular routing, they may actually be slower and are certainly more difficult to route.

Adder structure increases for increased bit multiplication

Volume 4 Issue III, March 2016
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

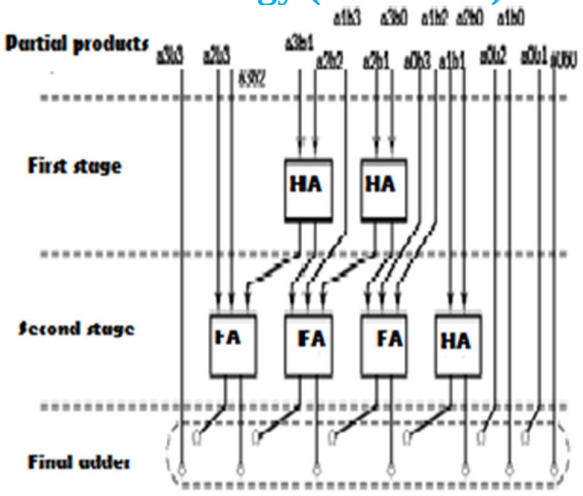


Fig.7. Wallace tree multiplier

B. Design Approach

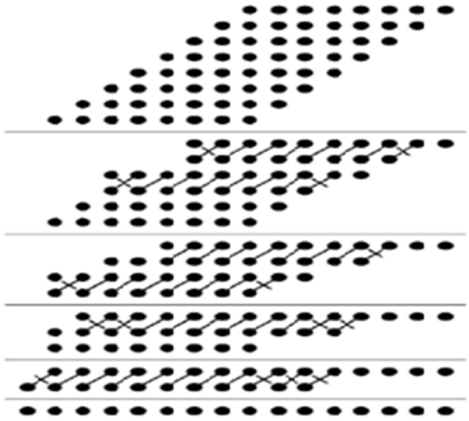


Fig.8. 8 bit conventional Wallace tree

The number of partial products is determined in each stage is determined by the following equation:

$$w_0 = N$$

$$w_{j+1} = 2.\, w_j \,/\, 3 + w_j \bmod 3 \qquad (10)$$

C. Modified Wallace Tree

In this multiplier we first form an inverted pyramid array by rearranging the bits in the partial product array

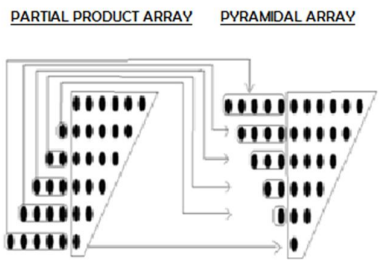


Fig.9. conversion of partial product array to pyramidal array

The next step is partial product reduction. This is done again by grouping the partial product array into 3 rows. In this case we

don't use the half adders as we did for conventional Wallace tree. So the number of half adders is reduced. Here we use half adders whenever the number of rows in each stage exceeds that of the previous stage.

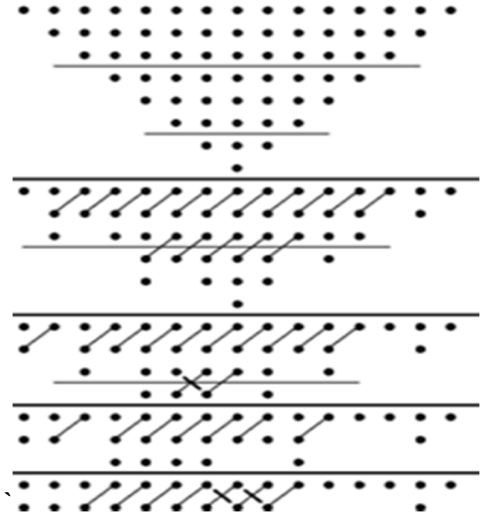


Fig.10. 8 bit modified Wallace tree

IV. SIMULATION AND ANALYSIS

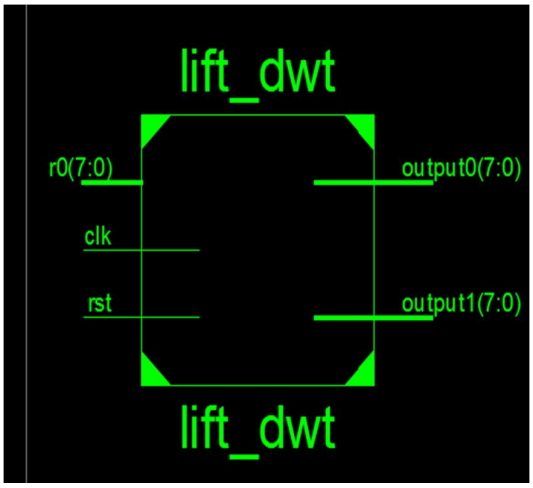


Fig.11.rtl schematic of lifting based DWT

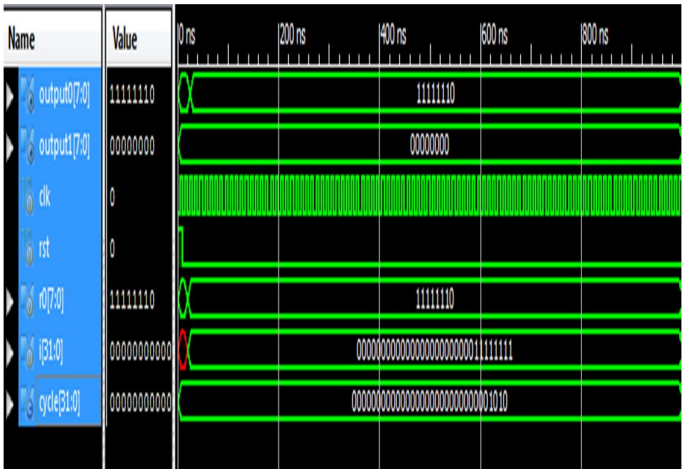


Fig.12. Simulated output of lifting based DWT

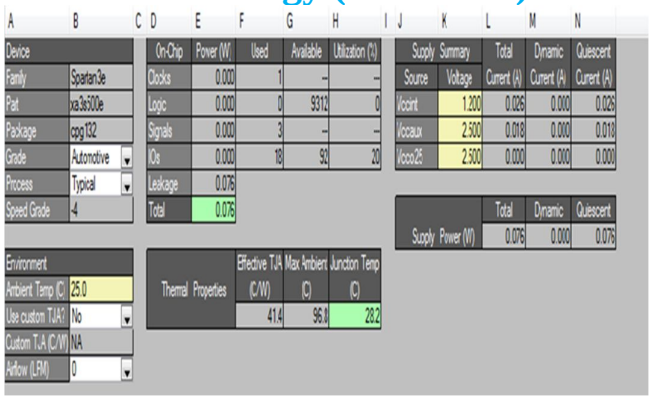


Fig.13. power consumption by the multiplier

V. CONCLUSION

In this paper, we have analyzed the existing Lifting based 2-dimensional Discrete Wavelet Transform based on the hardware complexities and computational time for the different architectures using Lifting schemes. This review is useful for exploring a new method of pipelined architectures capable of handling multiple data streams suitable for application in image and video processing multimedia real time applications.

It is seen that modified Wallace tree has the least area and power. Hence the modified Wallace tree is used for 2D-DWT. The DWT is also used for signal processing applications. 2D-DWT uses low pass and high pass filters for signal analysis and reconstruction. The area, power and delay parameters are also found out using Synopsys design compiler and Xilinx tools. The main challenges in the hardware architectures for 1-D DWT are the processing speed and the number of multipliers and adders while for 2-D DWT it is the memory issue that dominates the hardware cost and the architectural complexity. It has been tackled while a multiplier is designed using the lifting scheme.

VI. FUTURE SCOPE

Future work or the application can be done using the Vedic multiplier. The power, area, and delay can be calculated for Wallace tree multiplier and Vedic multiplier. The comparison of these parameters will help to understand which multiplier is well suited for the lifting based DWT. The hardware implementations can be made easier using these multipliers.

REFERENCES

[1] Mrs. Geetha K “Approximation Method for High Speed Multiplier-Less Dwt Architecture”IOSR Journal of Electronics and Communication Engineering (IOSRJECE) ISSN : 2278-2834 Volume 1, Issue 6 (July-Aug 2012), PP 12-18

[2] Chao Cheng, Member, IEEE, and Keshab K. Parhi, Fellow, IEEE “High-Speed VLSI Implementation of 2-D Discrete Wavelet Transform” IEEE transactions on signal processing, vol. 56, no. 1, January 2008

[3] Ibrahim Saeed and Herman Agustiawan “Lifting-based VLSI Architectures for Two-Dimensional Discrete Wavelet Transform for Effective Image Compression” Proceedings of the International MultiConference of Engineers and Computer Scientists 2008 Vol I IMECS 2008, 19-21 March, 2008, Hong Kong

[4] Athira Koranath, Sonali Agrawal “Comparison of different multiplier algorithms and 1D-DWT as an application”IOSR Journal of Electronics and Communication Engineering (IOSRJECE) ISSN : 2278-2834 Volume 1, Issue 1 (May-June 2012), PP 43-48

[5] Maria E angelopaulo and Peter Y. K. Cheung “Implementation and Comparison of the 5/3 Lifting 2D Discrete Wavelet Transform Computation Schedules on FPGAs” Journal of VLSI Signal Processing 2007

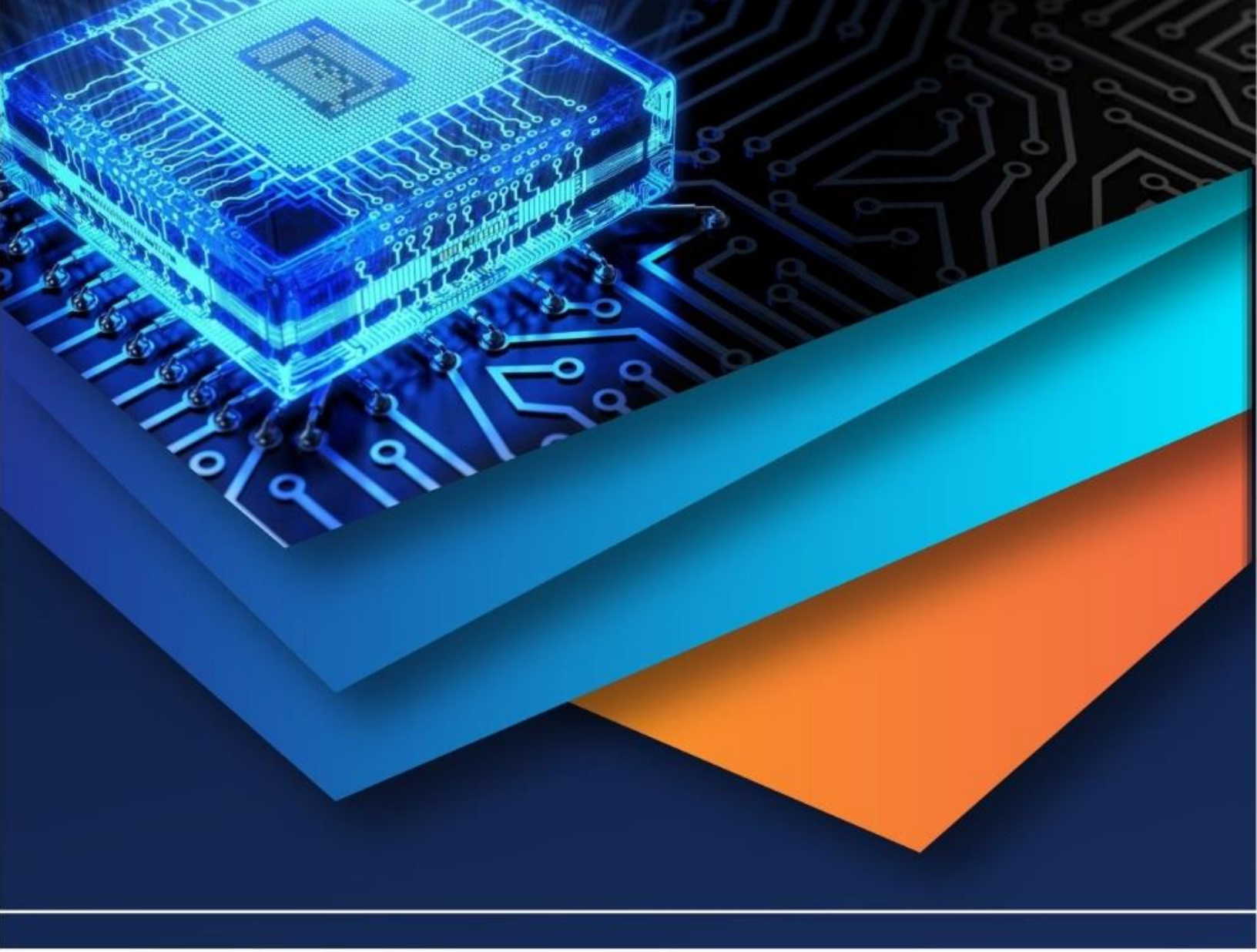
[6] Durga Sowjanya1, K N H Srinivas2 and P VenkataGanapathi “FPGA implementation of efficient VLSIarchitecture for fixed point 1-d dwt using lifting scheme” International Journal of VLSI design & Communication Systems (VLSICS) Vol.3, No.4, August 2012

[7] JoshniC.George, T.Jayachandran, Dr.C.N.Marimuthu “2D-DWT lifting based implementation using VLSI architecture”International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE) Volume 2, Issue 3, March 2013

[8] M. Janardan, Dr. K Ashok Babu “An efficient architecture for 3-d lifting-based discrete wavelet transform” M Janardan et al, Int. J. Comp. Tech. Appl., Vol 2 (5), 1439-1458

[9] P.C. Wu and L.G. Chen ,(2001), “An efficient architecture for two-dimensional discrete wavelet transform”,IEEE Transaction on Circuit and Systems and Systems for Video Technology, Volume 11,No. 4, Pages 536-545.

[10] MiladGhantous, MagdyBayoumi, (2011), “P2E-DWT: A Parallel and Pipelined Efficient VLSIArchitecture of 2-D Discrete Wavelet Transform”,.IEEE.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)