High performance RNS based DWPT for Image processing application

D. Vijaya Sree (Roll No: 2451-20-744-003), Dr. D.R.V.A Sharath Kumar (Associate Professor)

Department of Electronics and Communication Engineering

Maturi Venkata Subba Rao (MVSR) Engineering College

Saroornagar Mandal, Badangpet - Nadargul Main Rd, Hyderabad, Telangana 501510

***Abstract*— The lifting-based wavelet computing paradigm emerged as a key alternative for trading off accuracy and energy efficiency. The lifting based Discrete Wavelet Transform (DWT) pipelined VLSI architecture with an arbitrary wavelet tree is used. Here memory utilization rate is optimally reduced using bypass and reordering models. The lifting discrete wavelet transform (LDWT) is a low-complexity pre-processing filter suitable for image processing systems which are incredibly energy constrained. This work proposes the replacement of conventional arithmetic with improved RNS arithmetic model in a DWT hardware architecture and also to implement this by using Image processing application. The use of this technique helps to improve the system performance in terms of path delay and energy efficiency which validates the performance metrics using image quality assessment (IQA) measurements. In summary: the results prove the energy and delay reduction with approximation and the highest acceptable compression ratio processing systems which are incredibly energy constrained**

***Index Terms*— approximate computing, discrete wavelet transform (DWT), VLSI design, RNS system, image processing.**

I INTRODUCTION:

The discrete wavelet packet transform (DWPT) gives considerably more adaptable time and recurrence disintegration and ends up being an effective technique for the inadequate portrayal of a wide class of signals. The disintegration of both the approximation and definite coefficients adds more subtleties at each level. It is widely used in multimedia and signal processing.

A few structures are proposed for the effective calculation of full or partial DWPT decomposition. DWT transformation involves decomposition of image into four sub bands. This involves steps like **Low frequency band** – Basic image information and **High frequency band** – Edge of sharpness of images. Multi-level decomposition decomposes each band into numerous sub bands.

In this brief, the proposed architecture incorporates RNS system which encodes a large number into a group of small numbers and results in significant speed up of the overall data processing. This system has been incorporated as an improved changed for the better performance in the hardware architecture of Lifting Discrete Wavelet Packet Transform with arbitrary tree structure. The image processing is the application that helps to implement the proposed design with improved delay and energy efficiency.

The rest of this brief is categorized as mentioned below. Section II describes the DWPT computation with an arbitrary tree. Section III describes the 3- level 8- point DWT Section IV and V gives brief about the RNS method incorporation and validation using Image processing. Finally, the conclusions aredrawn in Section V.

II DWT COMPUTATION WITH ARBITARY TREE USING LIFTING SCHEME:

A wavelet can be defined as a wave like oscillation that is localized in time. Here the main idea rests regarding the computation of wavelet. For a specific scale and particular location or placement of it.

Lifting scheme comes into the picture which was introduced by Swelden. This has been proved as an efficient method to compute DWT/DWPT. When it comes to the point of FIR wavelet filter the factorization is categorized into sequence of lifting steps. The Euclidian algorithm is used to factorize the polyphase matrix. This kind of factorization results in a number of upper and lower triangular matrices and a diagonal normalization matrix.

$$P(Z) = \begin{bmatrix} h_e(z) & g_e(z) \\ h_0(z) & g_o(z) \end{bmatrix}$$

$$= \prod_{i=1}^{m} \begin{bmatrix} 1 & s_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ t_i(z) & 1 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & 1/ \end{bmatrix}$$

Here, $si(z)$ and $ti(z)$ = Laurent polynomials

K = Constant

Computation of these upper triangular matrix gives highpass sub-band as output and the lower triangular matrix computation results in lowpass sub-band as output. Finally, the outputs are scaled by K and 1/K alternatively. This has proved to be efficient in reducing the computational complexity approximately by 50% when compared to that of the convolution method.

The main difference between the classical wavelet construction and that of the lifting scheme is that the lifting scheme does not rely on Fourier transform and is used to construct second-generation wavelets.

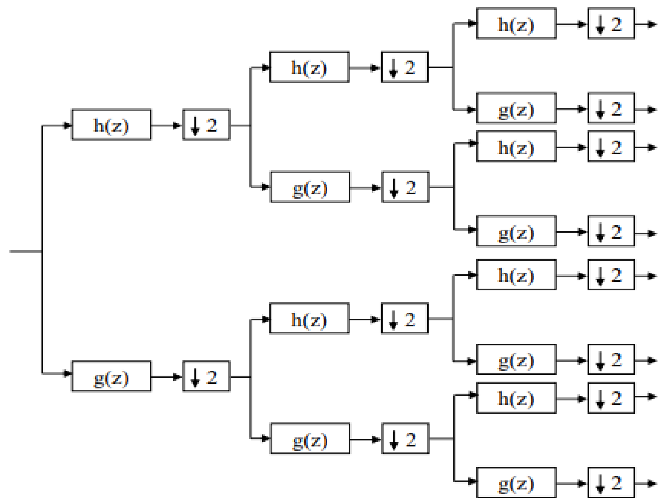


Fig.1 Level tree-structured filter bank.

One more advantage of using this scheme is that the number of filter co-efficients used are lesser when compared to that of eqivalent convolution filter and provides faster implementation. Sequence of operations involved in performing lifting scheme are:

Split a_j into Even-1 and Odd-1

$$d_{j-1} = odd_{j-1} - Predict(even_{j-1})$$
$$a_{j-1} = even_{j-1} - Update(odd_{j-1})$$

It consists of 3 main steps and they are:

1) Splitting.
2) Predict.
3) Update.

In **splitting** the initial data is further sampled into odd half and even half sets.

Predict helps to forecast the missing bits from Split, it creates a prediction operator, which is often based on a model of the signal data. The wavelet coefficients are determined by the failure to anticipate the odd set based on the even set. This particular step is also called as Dual Lifting and it is a high pass filter.

Update phase updates the even set using the wavelet coefficients already computed for the scaling function coefficients, ensuring that all wavelet coefficients at all levels have the required qualities. This is often referred to as the Lifting step.

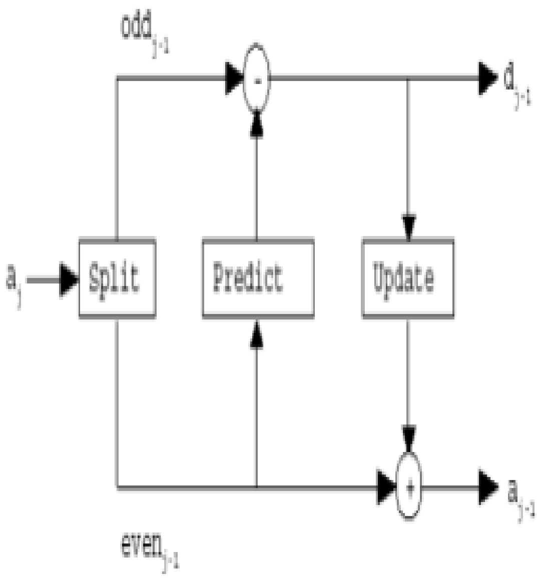


Fig.2 Lifting Scheme of forward wavelet transform

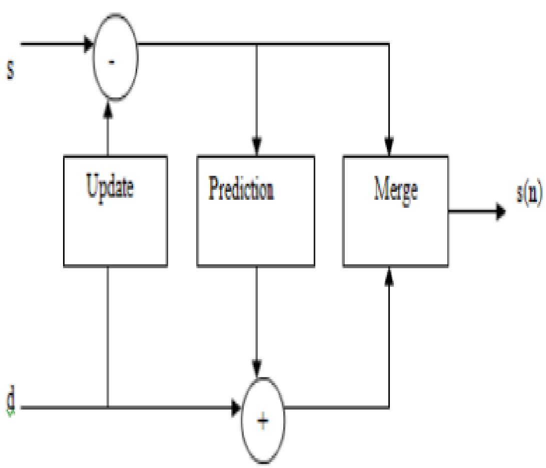


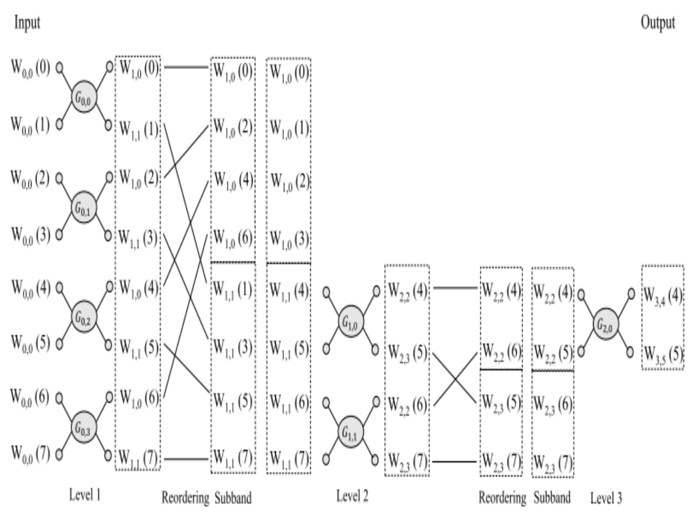
Fig.3 Lifting Scheme of inverse wavelet transform

III 3-LEVEL 8-POINT DWT/DWPT:

The DWPT coefficients are calculated in phases for different levels. Wavelet coefficients are generated and reordered to generate sub-bands at each level using the suggested wavelet filter. From a single path, data enters and departs the architecture serially. The top column in the table shows the time in the clock cycle, while the other columns show the input and output of each module.

Each column is further divided into two columns, one for the filter's input and the other for the reordered filter output.

Fig.4 SFG of 3-level 8-point full DWPT decomposition with arbitrary tree.



The final coefficients in Module 1 are the number of coefficients W1,0(t) computed in Module 0. Modules 1 and 2 bypass these coefficients, which are produced as final coefficients at the output end. Similarly, the final coefficients W2,3(6) and W2,3(7) calculated in Module 1 are obtained.

Bypassed Lifting-Based Wavelet Filter:

The lifting scheme with Euclidian algorithm for factorization of the polyphase matrix. The bypass registers are used in the predict and update stages, which are pipelined in order. The lifting-based filter suggested here computes detailed and estimated coefficients for serial data as an input sample.

IV VALIDATION USING IMAGE PROCESSING AND USING RNS METHOD:

Using an image processing application, the performance metrics of the proposed DWT multi-level decomposition model is tested and also demonstration of its energy compaction and band decomposition measures in the form of lower and upper bands.

RNS (RESIDUE NUMBER SYSTEM):

RNS is a technique that is used in the place of adders, subtractors, multipliers and all those that are involved in the calculation. For energy efficiency and path delay optimization, the RNS system is introduced. Parallel processing is feasible. There is a significant reduction in both area and path delay.

The overall data processing is significantly speed up by encoding a large number into a group of little integers. A set of moduli characterizes an RNS. The residues are a set of smaller integers that can be used to represent any large integer.

Steps involved:

Forward conversion
(number → residue's)
Arithmetic computation
(+,-,x)
Reverse converter
(residue's → number)

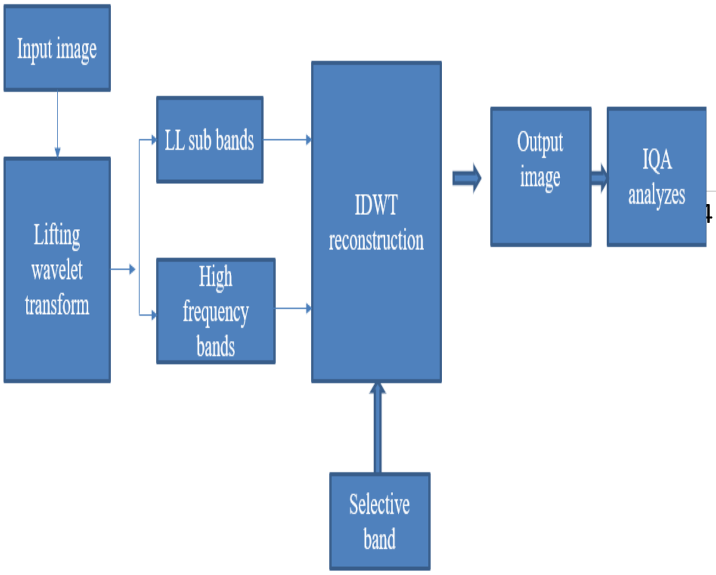


Fig.5 BLOCK DIAGRAM OF MODIFIED DESIGN

The input image is transformed to a digital file (.txt). The memory has been set up to load the input image. The Haar wavelet technique is used to create sub-band outputs (Low and high). For frequency band creation, use Muti level (3) decomposition. Inverse transform models for images.

Process of selective band breakdown. Measurements of image quality assessment (IQA).

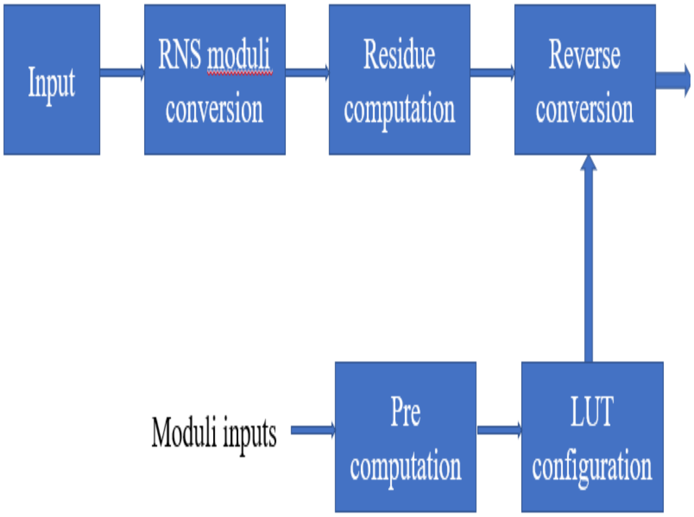


Fig.6 BLOCK DIAGRAM OF RNS SYSTEM

```
manual calculations - Notepad
File  Edit  Format  View  Help
A           B

Ax B

A ---  ra1 ra2 ra3
B ---  rb1 rb2 rb3    moduli conversion   (forward coversion)

m1m2m3  7 8 9


A B   13 20

A ---  ra1 ra2 ra3
13 --- 13%7 13%8 13%9
13 ---  6     5     4

B ---  rb1 rb2 rb3
20 --- 20%7 20%8 20%9
20 ---  6     4     2

therefoe, 13*20 = 260


residue computation

=r1= ra1.rb1 % 7 =1
r2 - ra2 .rb2%8 =4
r3=  ra3.rb3%9 =8



reverse conversoin
|
|MOUT= | r1.M1.M1`  + r2.M2.M2`+ r3.M3.M3` | m1m2m3|

m1 m2 m3= 789
M1= m2.m3
M2= m1.m3
M3= m1.m2

M1 M2 M3 = 72 63 56

M1.M1`  % 7 =1(assume and vary until we get the correct value)
M2.M2`  % 8 =1
M3.M3`  % 9 =1


M1` M2` M3`  ==   4  7  5


|MOUT= | r1.M1.M1`  + r2.M2.M2`+ r3.M3.M3` | m1m2m3|
    =   1.72. 4    +  4 .63. 7 + 8 . 56. 5 |504|
    =   4292 |504|
    =   260
```

Fig. 7 Manual Calculations

The input is an image of 8x8 macro blocks. In Moduli conversion the input will be converted into three residues each ra1, ra2, ra3 and rb1, rb2, rb3 and this is also called as forward conversion and assume numbers accordingly m1, m2, m3 and using the formula as shown residue computation will be done or obtained. Then comes the reverse conversion and using the formula the calculation will be carried out.

Finally, from the manual calculations we are showing that the initial multiplication of two numbers 13x20=260. Whereas, after applying the RNS method we got the exact result of 260 at the end as output at |MOUT|.

V IMPLEMENTATION USING IMAGE PROCESSING:

In this section, we'll look at how to use MATLAB to create 2D FDWT and IDWT with thresholding. The input data will be converted from the time domain to the scale domain in the FDWT section. Then, in the thresholding section, some of the coefficients will be set to zero, and the coefficients will be moved back into the time domain in the IDWT section. The matrix multiplication approach was utilized to implement the algorithm in MATLAB. For MATLAB simulation, we used the picture input file, as well as 8 randomly chosen image co-efficients. In several ways, the construction of algorithms in Verilog HDL differs. The key distinction is that, unlike MATLAB, Verilog HDL lacks numerous built-in functions like convolution, max, mod, flip, and others. FDWT and IDWT linear equations are utilized to build the technique in Verilog HDL.

VI RESULTS:

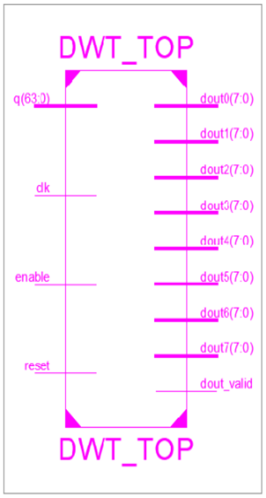


Fig.8 RTL Schematic

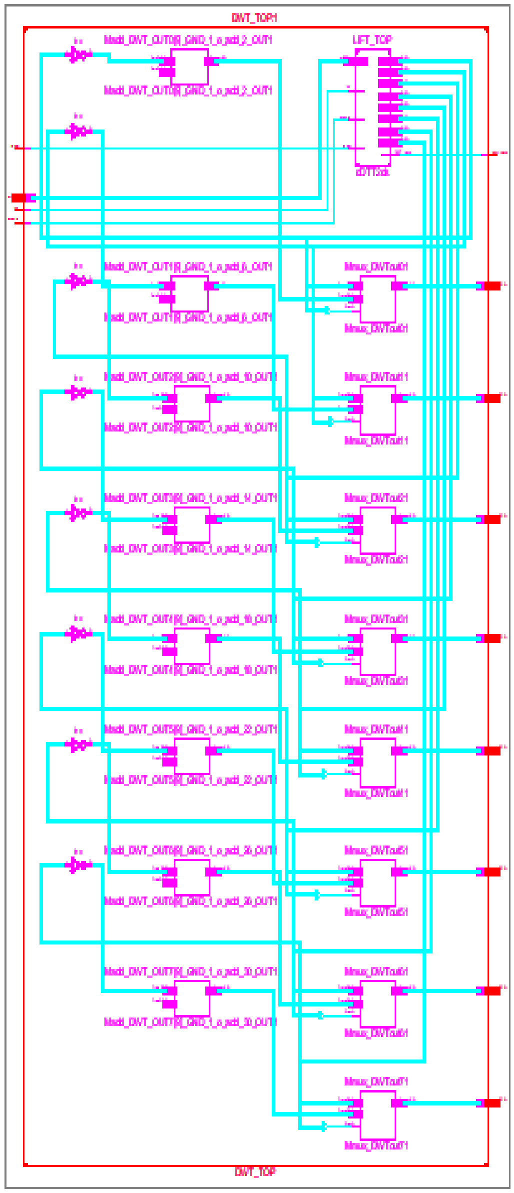
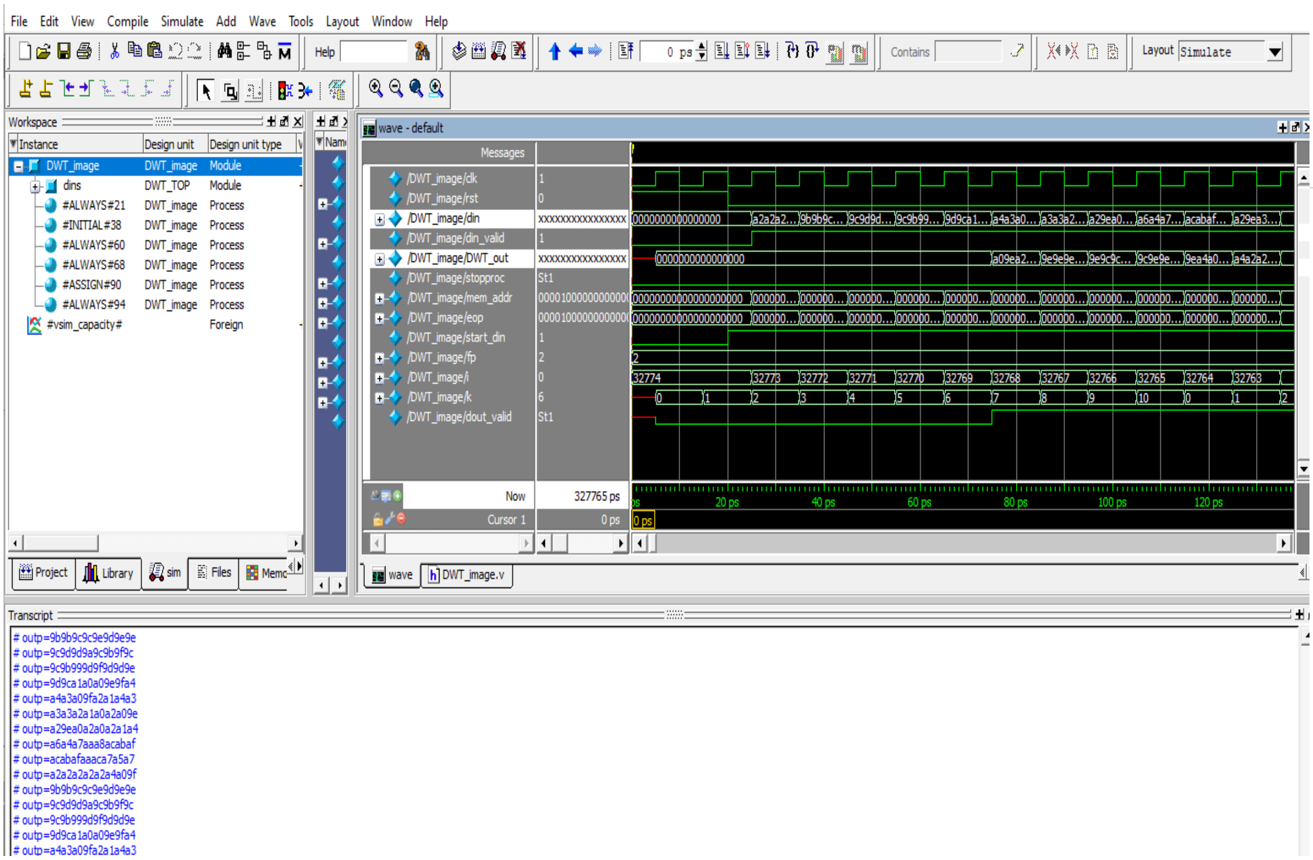


Fig.9 Technology Schematic



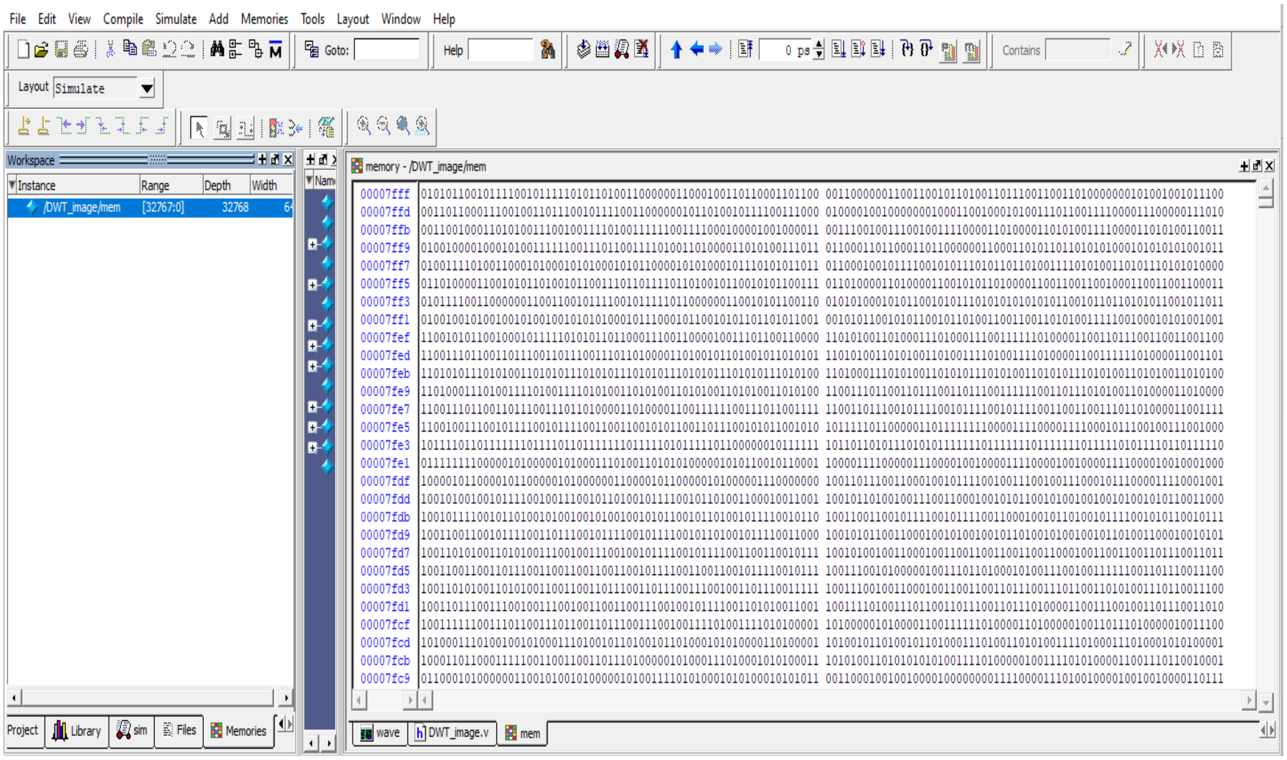
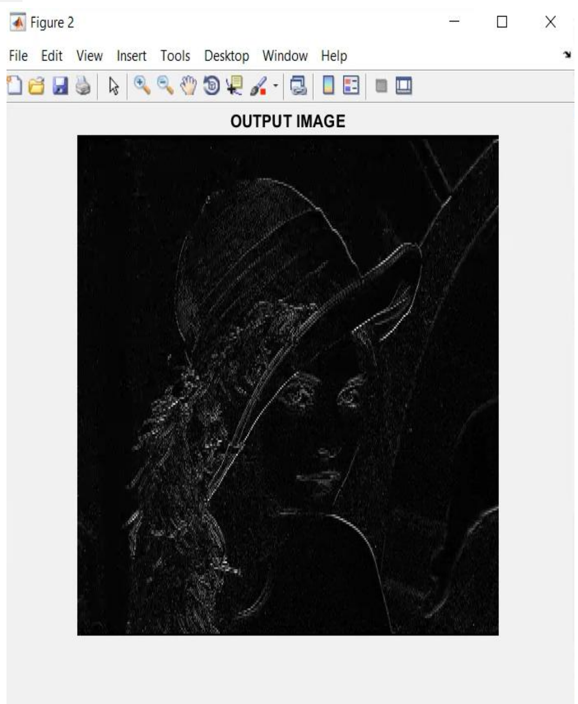


Fig.10 Simulation Results

Image processing Results



VII REFERENCES:

[1] M. M. Hasan and K. A. Wahid, "Low-cost lifting architecture and lossless implementation of Daubechies-8 wavelets," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 65, no. 8, pp. 2515–2523, Aug. 2018.

[2] Gyanendra Singh, Samba Raju Chiluveru, Balasubramanian Raman, Manoj Tripathy, and Brajesh Kumar Kaushik, "Novel Architecture for Lifting Discrete Wavelet Packet Transform with Arbitrary Tree Structure", IEEE Trans. Very Large Scale Integration (VLSI) systems, vol. 29, no. 7, July 2021.

[3] M. M. Hasan and K. A. Wahid, "Low-cost architecture of modified Daubechies lifting wavelets using integer polynomial mapping," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 64, no. 5, pp. 585–589, May 2017.

[4] S. K. N. Mahammad, "An efficient VLSI architecture for lifting based 1D/2D discrete wavelet transform," Microprocessors Microsyst., vol. 47, pp. 404–418, Nov. 2016.

[5] D. Garcia, M. Mansour, and M. Ali, "A flexible hardware architecture for wavelet packet transform with arbitrary tree structure," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 60, no. 10, pp. 657–661, Oct. 2013.

[6] W. Zhang, Z. Jiang, Z. Gao, and Y. Liu, "An efficient VLSI architecture for lifting-based discrete wavelet transform," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 59, no. 3, pp. 158–162, Mar. 2012.

[7] M. Garrido, J. Grajal, and O. Gustafsson, "Optimum circuits for bit reversal," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 58, no. 10, pp. 657–661, Oct. 2011.

[8] C. Wang and W. S. Gan, "Efficient VLSI architecture for lifting-based discrete wavelet packet transform," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 54, no. 5, pp. 422–426, May 2s007.

[9] G. Shi, W. Liu, L. Zhang, and F. Li, "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.

[10] H. Liao, M. K. Mandal, and B. F. Cockburn, "Efficient architectures for 1-D and 2-D lifting-based wavelet transforms," IEEE Trans. Signal Process., vol. 52, no. 5, pp. 1315–1326, May 2004.

[11] 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.

[12] M. Jou, Y. H. Shiau, and C. C. Liu, "Efficient VLSI architectures for the biorthogonal wavelet transform by filter bank and lifting scheme," in Proc. IEEE Int. Symp. Circuits Syst., vol. 2. Sydney, NSW, Australia, May 2001, pp. 529–532.

[13] C.-J. Lian, K.-F. Chen, H.-H. Chen, and L.-G. Chen, "Lifting based discrete wavelet transform architecture for JPEG2000," in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), vol. 2, Sydney, NSW, Australia, May 2000, pp. 445–448.

[14] 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.

[15] C. Chakrabarti and M. Vishwanath, "Efficient realizations of the discrete and continuous wavelet transforms: From single chip implementations to mappings on SIMD array computers," IEEE Trans. Signal Process., vol. 43, no. 3, pp. 759–771, Mar. 1995.