A Computationally Efficient Pipelined Architecture for 1D/2D Lifting Based Forward and Inverse Discrete Wavelet Transform for CDF 5/3 Filter

Serap CEKLI
Maltepe University, Computer Engineering Department, Istanbul, Turkey
serapcekli@maltepe.edu.tr

***Abstract*—In this study, a simple lifting based pipeline DWT (Discrete Wavelet Transform) architecture is proposed for the operation of the CDF 5/3 (Cohen-Daubechies-Feauveau 5/3) filter. This scalable architecture is faster and capable of fulfilling the transformation utilizing the parallel processing operation units. The symmetric boundary extension method is used at the signal boundaries to obtain the best result in the case of 1D/2D. The proposed architecture utilizes the hardware resources in a quite efficient way by means of the pipeline technique. The architectural design is constituted by using RTL (Register Transfer Level) design process and coded by the Verilog HDL. The proposed architecture is tested for several 1D/2D inputs to examine its operation. The related architecture is synthesized for the FPGA board to check the results. The reverse operation is fulfilled by using the same structure only by changing the shift amounts of the shifting units. The DWT coefficients are calculated on this architecture for the 1D/2D situation. The hardware resources are used effectively by utilizing the constituted architecture in folded structure in the 2D case. Satisfying results have been obtained when the different numbers of parallel processing units are utilized.**

***Index Terms*—digital systems, discrete wavelet transforms, multiprocessing systems, pipeline processing, programmable logic arrays.**

I. Introduction

The wavelet analysis which expands the influence area continuously has been an important tool since it has been first introduced for the multiresolution signal decomposition [1]. The wavelet transform is a very convenient and effective transform at almost for all kind of application fields including the JPEG compression algorithms of the signal and image processing studies [2-7]. The wavelet analysis is a versatile, efficient and helpful technique in biomedical applications [8-10], genomic researches [11-13], and even seismic signal processing [14-15], as well in the studies of the other research fields [16] for the researchers. The power of the wavelet transform stems from providing the better localization in time-frequency domains.

The applications and the designs of the various studies are implemented by using digital systems which have limited throughput and memory capacity. Therefore, the efficient realization of the Discrete Wavelet Transform (DWT) which provides the sufficient information to analyze and synthesize the original signal is considerably crucial. The resulted coefficients from this computationally intensive signal transform should be calculated by using an effective digital architecture with satisfying the requirements and characteristics of the related application. So far, several architectures and VLSI (Very Large Scale Integration) systems are proposed for this purpose [4], [17].

The filter banks, which are composed of lowpass, and highpass filters essentially, are used for the wavelet transform, and these filter banks are implemented with different methods. These are convolutional based approaches including direct form, cascade, polyphase, lattice structure, and lifting based structure [3-4], [18-20]. Each of these convolutional based methods has some advantages and disadvantages depending on the application fields. However, a systematic and useful method for the biorthogonal wavelet transform has been proposed. Lifting method for wavelet transform is the spatial constructions of the wavelet coefficients by deriving the polyphase matrices [21-22]. This reversible factorization technique consisting of the lifting steps is preferred substantially and encountered in literature widely due to the its attractive properties such as being a transform of providing integer to integer mapping, in-place processing, and spatial instead of frequency [3], [20], [23-24].

The lifting method is a filter bank which is utilized to compute the DWT coefficients using biorthogonal wavelets. Some of the studies are based on the pipeline technique [25-26], some of them uses the folded architecture in which the use of the single filter output of multiple times for the n levels [27-28]. Some others utilize the flipping structure [29-30]. Each of these architectures has purposed some improvements considering some different performance criteria just as hardware complexity, memory requirements, precision, throughput and latency.

In this study, a simple and efficient architecture is proposed to realize the lifting based DWT method which is used in the various studies including today's important applications such as image processing and JPEG compression [31]. So, the parallel processing elements, which are easily affordable today in terms of cost, are used in the structure of this 2D pipelined DWT architecture which is benefiting from the power of the parallel operation and increasing the system performance. Some efforts in this point of view are appeared in several studies [32]. The proposed scalable architecture can be useful for the vector processors with an appropriate adaptation [33].

Digital Object Identifier 10.4316/AECE.2018.02003

Another crucial issue is that the computation of the 2D image borders, in other words the decision about the method which is used for the 2D image border extension is affected the transform performance at the image borders significantly. There are some available studies regarding to this critical issue in the literature [34-35]. Proper one of the boundary extension methods, including symmetrical, zero padding, periodic extension, can be used for the computation of the boundaries of the signal by considering the limitations and the necessities of the related application. In this study, the symmetric boundary extension method is employed to have the optimum performance and not to increase the hardware complexity.

An architecture for the CDF 5/3 (Cohen-Daubechies-Feauveau) filter which is important for the lossless JPEG compression is proposed. The application of the shifting units is sufficient instead of the use of multipliers by taking the advantage of the dyadic nature of the 5/3 CDF filter.

The FPGA (Field Programmable Gate Arrays) boards are used to test and examine the designed architectures in literature frequently [17], [37]. The proposed architecture has been formed using RTL (Register Transfer Level) design process. The RTL design process is used for the algorithmic level design describing the signal flows between the registers of the complex or large systems synchronized by a clock signal. So, the proposed design which is obtained by using the RTL design process has been verified by simulation and synthesized for the Xilinx Spartan 3e FPGA board to examine the results. Also, Verilog HDL (Hardware Description Language) has been used to describe of the designed architecture.

For the 5/3 CDF filter, reconstruction of the wavelet coefficients from the original signal is lossless. Because 5/3 CDF filter coefficients are dyadic rational values and so the integer to integer transform is obtained by using this feature [38]. The implementations with finite precision should be constructed when the wavelet filters with irrational coefficients are used, and this situation leads to the data loss. However, the original samples are obtained even though the fractions of the transformed wavelet coefficients are rounded to the integers when the 5/3 CDF filter is used. The 5/3 CDF filter is known as lossless wavelet transform, and also the JPEG 2000 standard which is the compression standard using wavelet method supports the lifting based filtering mode. Also, this filter belongs to the two-matrix lifting factorization class.

So, in this study an efficient architecture is proposed for the widely used CDF 5/3 filter on account of the properties of the lossless wavelet transform. The realization of the inverse transform is also required to test the performance of the proposed architecture.

The rest of the paper is organized as follows. The background of the wavelet transform is summarized as looking from the viewpoint of lifting schema in the second section. The essential constructs of the proposed 2D DWT processors as well as the general system perspective are described in the third section. The detailed explanation of the proposed 1D/2D lifting based architecture is also given. After that the results and discussions are given, and finally conclusion section is given.

II. Wavelet Transform Background And Lifting Schema

DWT uses two sets of functions. These functions are the scaling functions relating with lowpass filter, and the wavelet functions relating with highpass filter.

Lifting schema is a convenient method to compute DWT using biorthogonal wavelets. Lifting method can be used if there is no appropriate one among the known wavelets for the application under consideration [38].

The information carried on a signal does not vary from sample to sample, randomly. There is a correlation between the adjacent samples. The even indexed samples can be predicted by using only the odd indexed samples of the data at hand, and vice versa. The accuracy of the prediction depends on the correlation between the adjacent samples and the suitability of the estimation method [21], [39].

Lifting factorization method, also called the second generation wavelets, is consisted of three steps. These steps can be summarized as follows;

I) Split: The signal is separated into its even and odd polyphase components. This operation is also called the Lazy wavelet. The even indexed samples are the even polyphase components and the odd indexed samples are the odd polyphase components of the signal.

II) Predict: The odd samples are predicted using the adjacent even samples at this step. The detail coefficients in other words highpass components are computed.

III) Update: The even samples are computed by means of the odd samples obtained at the previous step. Indeed, the approximation coefficients, also called lowpass components, are computed at this step [39].

In brief, the new odd samples are predicted using even samples and the new even samples are updated using the new odd samples. The z-transform of a FIR filter, which is a delay line essentially, can be represented by Laurent series.

$$h(z)=\sum_{k=p}^{q} h_k z^{-k} \tag{1}$$

where h_k is impulse response of a filter h which has the finite number of coefficients h_k, and the degree of that Laurent polynomial h is given as $abs(h)=q-p$.

The polyphase matrix used for the lifting schema is defined as:

$$P(z)=\begin{bmatrix} h_e(z) & g_e(z) \\ h_o(z) & g_o(z) \end{bmatrix} \tag{2}$$

where $h_e(z)$ defines even low pass coefficients, $h_o(z)$ defines odd low pass coefficients, $g_e(z)$ defines even indexed high pass coefficients and $g_o(z)$ defines odd high pass coefficients.

The related lifting and scaling steps are derived from the biorthogonal wavelets to realize the lifting based DWT. The analysis filters of a specific wavelet class should be represented in the polyphase matrix form. The polyphase matrix of a wavelet transform filter is decomposed into the upper and lower triangle matrix, and the diagonal matrix. Thus, the lifting based architectures are derived.

The $P(z)$ matrix having the determinant value of 1 is required to perform the wavelet transform. In other words, the diagonals of the polyphase matrix are 1, and the matrix is factorized as 2x2 upper and lower triangle matrix. The

upper triangle matrix defines the prediction step coefficients, and the lower triangle matrix defines the update step coefficients.

The (*h, g*) is complementary filter pair, Laurent polynomials are $s_i(z)$ and $t_i(z)$ for $1 <= i <= m$, and K is a non-zero constant to factorize the polyphase matrix.

$$P(z)=\prod_{i=1}^{m}\begin{bmatrix}1 & s_i(z)\\ 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0\\ t_i(z) & 1\end{bmatrix}\begin{bmatrix}K & 0\\ 0 & 1/K\end{bmatrix} \quad (3)$$

The polyphase matrix can be factorized into lifting steps routinely. Each finite filter used for the wavelet transform is obtained by utilizing *m* times lifting and dual lifting steps following the Lazy wavelet step. The last one of these steps is the scaling. In the scaling step, the scaling matrix having each element is 0 except for the diagonal ones is derived. The dual polyphase matrix is given as,

$$\tilde{P}(z)=\prod_{i=1}^{m}\begin{bmatrix}1 & 0\\ -s_i(z^{-1}) & 1\end{bmatrix}\begin{bmatrix}1 & -t_i(z^{-1})\\ 0 & 1\end{bmatrix}\begin{bmatrix}1/K & 0\\ 0 & K\end{bmatrix} \quad (4)$$

The reverse operation of lifting schema is obtained by alternating the signs of the Laurent polynomials, the sign of the $s_i(z)$ and $t_i(z)$ and running the all operations backward direction [23]. The lifting schema is a suitable alternative when the frequency based techniques are not practical.

A. The 2D lifting schema based DWT

The CDF 5/3 filter is a reversible transform valid for the JPEG 2000 image compression algorithm and coding standard for lossless image compression. The filter coefficients are dyadic rationals which can map integer to integer. The highpass and lowpass coefficients through the transform are the complements of the each other, and the original input signal can be reconstructed by means of these two time series together. The coefficients of this lossless transform filter are given as follows;

Lowpass: (-1/2, 1, -1/2)

Highpass: (-1/8, 2/8, 6/8, 2/8, -1/8)

The polyphase matrix and factorization of the filter in z domain can be given in the following form;

$$P(z)=\begin{bmatrix}1 & 0.25(1+z)\\ 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0\\ -0.5(1+z^{-1}) & 1\end{bmatrix} \quad (5)$$

The prediction and update steps of the CDF 5/3 filter are shown in the factorization given in the Eq. (5). Accordingly, the calculated new coefficients of the transform i.e. the odd and even components corresponding to the highpass and lowpass components in time domain are given as following equations,

$$y(2i+1)=x(2i+1)-0.5x\left(x(2i)+x(2i+2)\right) \quad (6)$$

$$y(2i)=x(2i)+0.25x\left(y(2i+1)+x(2i+3)\right) \quad (7)$$

where *x*'s are the signal points and *y*'s are the calculated signal samples [20].

III. General Structure Of The Proposed System

At first, the main system architecture is constructed. The word length has been chosen as 16-bit in accordance with the architecture in the system. The system composed of the main datapath, control and memory units. The system design has been performed by using RTL design process. The RTL design process is used for the algorithmic level design describing the signal flows between the registers of the complex or large systems. The ASMD (Algorithm State Machine Diagram) summarizing all the processes relating to the design and processor is constructed. The processes are executed depending on this designed chart.

The signal input starts sample by sample, as soon as the processor becomes ready by the defined enable input, i.e. start control signal, in the system. The number of signal sample points is N. The index of the first sample is 0 (even index). The even indexes show the even samples and odd indexes are indicating the odd indexed ones. The CDF 5/3 filter needs adjacent 3 samples of the input signal due to its characteristics. The first 3 samples are received cycle by cycle a total of 3 cycles, and the transform process starts immediately when these 3 samples acquired.

Initially, the new odd signal sample shown by the index "1" is computed. After the calculation of the "1" indexed odd sample, at this very moment the "0" indexed even sample is calculated. The symmetric boundary extension is utilized in order to avoid the performance degradation at the signal boundaries in the case of 2D transform when the first new even sample is computed. In this case, a copy of the "1" indexed new odd sample which is hold by a register is used for the calculation of the "0" indexed new even sample. Indeed, the register value containing the new calculated odd sample is used twice, in that situation. The pipeline technique is utilized at the datapath unit to provide a performance increase during the calculations. Considering the lifting based method, the pipelined technique appears to be a natural choice so the lifting itself is quite favorable.

The three successive samples are required to calculate the value of a new sample point, whether the odd or even, when regarding receive of the signal points into the system or more specifically the memory unit (or buffer). Therefore, taking the signal samples as pairs (odd-even or 3 indexed-4 indexed samples) is a necessity due to the data dependency. However, the time delay is prevented by means of the pipelined technique and an even new sample is calculated in each clock cycle, and an odd new sample is calculated in the other cycle, and so on. The datapath takes the "3" indexed odd sample while the "1" indexed new odd sample is computed.

The "4" indexed even sample is taken while the "0" indexed new even sample is computed. Two very next neighboring samples are necessary as well as the sample which just now under computation because of the data dependency. The pipeline technique is quite useful at this point, so this technique has been utilized in the proposed architecture to gain performance.

The pipeline technique is realized by using the extra pipeline registers. So, the hardware utilization is enhanced by means of some additional pipeline registers (or buffer registers) providing that there is no hardware resource is idle. The accurate input data is processed and transferred to the proper hardware resource by virtue of the control signals. The datapath operations are triggered by the clock signal, and total number of clock cycles of process time for input signal defines the total latency.

The process continues until the samples are finished. The symmetric extension is employed at the signal boundary using the copy of the "(N-2)" indexed sample when the

processor reaches the boundary. The process time schedule is shown in the Fig. 1. In the Fig. 1 solid lines; stands for the unchanged samples and dashed lines stands for the new calculated samples.

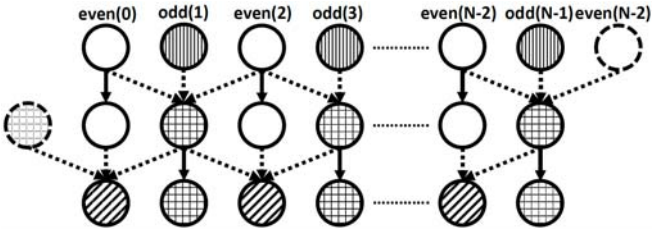

Figure 1. Data dependency diagram for even-odd signal sample points for the CDF 5/3 filter. It is clearly seen that the 3 adjacent samples are necessary in the case of the CDF 5/3 filter used. The first three samples are employed to compute “0” indexed new even. Last even (N-2) sample is placed for symmetric boundary extension

A. Proposed 1D lifting based architecture

The construction of the 1-D DWT architecture is the first step to establish the 2D DWT architecture because the 2D architecture is based on the 1D structure. For this purpose, the 1D architecture has been formed primarily. The architecture has been constructed by RTL design methodology and the model of the architecture is given in the Fig. 2. The architecture has been composed of the datapath unit, control unit and the memory unit essentially. An address unit is also added to organize and direct the addresses of the memory unit. The datapath executes all required operations, and the timing issues which are occurring in the datapath unit are treated by the control signals generated by the control unit.

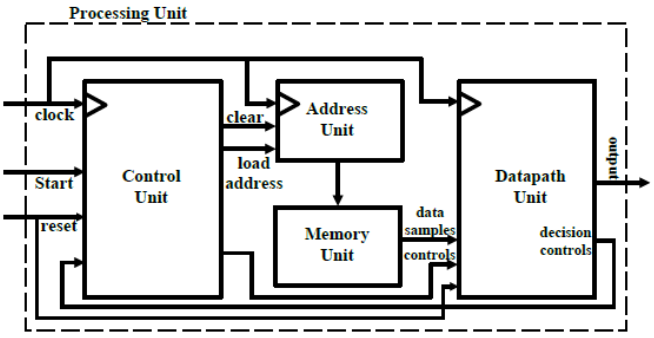

Figure 2. Model of processing unit which is designed for the CDF 5/3 filter architecture. Components which form the processing unit are shown

The pipeline technique provides the efficiency by reducing the total latency, and this technique is an effective choice for the proposed lifting based DWT. The registers hold the 16-bit fixed point numbers resulting from the 16-bit architecture. Realization of the DWT which utilize the FP (Floating Point) number format requires excessive hardware resource and moreover the use of the FP units is expensive solution. In addition, FP operations are slower as a consequence of the operation complexity. Low cost embedded microprocessors and microcontrollers do not have FPUs (Floating Point Units) generally; hence these systems do not support the FP (DSPs, FPGAs, etc.) operations.

These systems support the utilization of the fixed point format instead of using the FP number format. Especially the fixed point format is used for the FIR filters. Fixed point format provides the limited precision, but this format is more simple and faster to operate. Another advantage is that the fixed point systems require less hardware resources.

A synthesizable HDL code should be constituted to realize the architecture after design step. The synthesizable structures specify that the digital logic elements inside the FPGA board can implement the system design which is under consideration. Actually, synthesize operation is the conversion of the HDL code into circuit or hardware elements. The RTL design is quite convenient tool for the special purpose processor design just like in this study. The designed operations are executed in the datapath with the help of the control unit at each clock cycle. The design has some predefined specific properties which are summarized by the ASMD (Algorithm State Machine Diagram) given in the Fig. 3. Each state of the processor and the datapath operations corresponding to these states are described on the ASMD, distinctly. The control mechanism of the processor and pipeline registers are shown on the chart.

Datapath operations have been enumerated for convenience at the ASMD chart, and related operations are given from (1) to (11). The used decision control signals which are generated by datapath unit are given in Table I and the related datapath operations are given in Table II.

In given ASMD chart, the interactions between the datapath unit, control unit and memory unit are described. These operations are carried out at each processing unit simultaneously. Datapath operations could be summarized as follows. The image has been split into frames by address unit, and these frames are directed to corresponding processing units to have been filtered. When Start signal is asserted, datapath operations get starts, otherwise the control flow returns to "S_ready" state. "load_first_samples" control signal provides initialization of the sample counters. "S_first_there" state acquires the samples which are placed at the image boundaries. "Flush" signal flushes all the datapath operations and registers, if this signal is asserted upon necessity.

After having first three samples, the lifting based operations starts. One of the required control signals is asserted at each cycle, and the required pipeline stages are performed. A new sample point is taken to related register and at the same time sample counter is incremented. In that clock cycle, the new odd sample value is also calculated, and all these operations fulfilled upon activation of "Calculate1" signal. This stage is corresponding to the Eq. (6) i.e. the highpass components of the signal.

Using the previously calculated new odd sample values new even sample value is calculated similar to the previous step. This stage is corresponding to the Eq. (7) i.e. the lowpass components of the input signal upon activation of "Calculate2". After having these boundary samples then all other signal points are calculated by the help of the "Load1" and "Load2" control signals, until the sample number counter reach at the related number of sample count, which is given by N. When the sample counter reach at the related number of sample point, N, and then the "Load3" signal is activated and the other boundary samples of the image are calculated. All operations are pipelined using required pipeline registers and control signals.

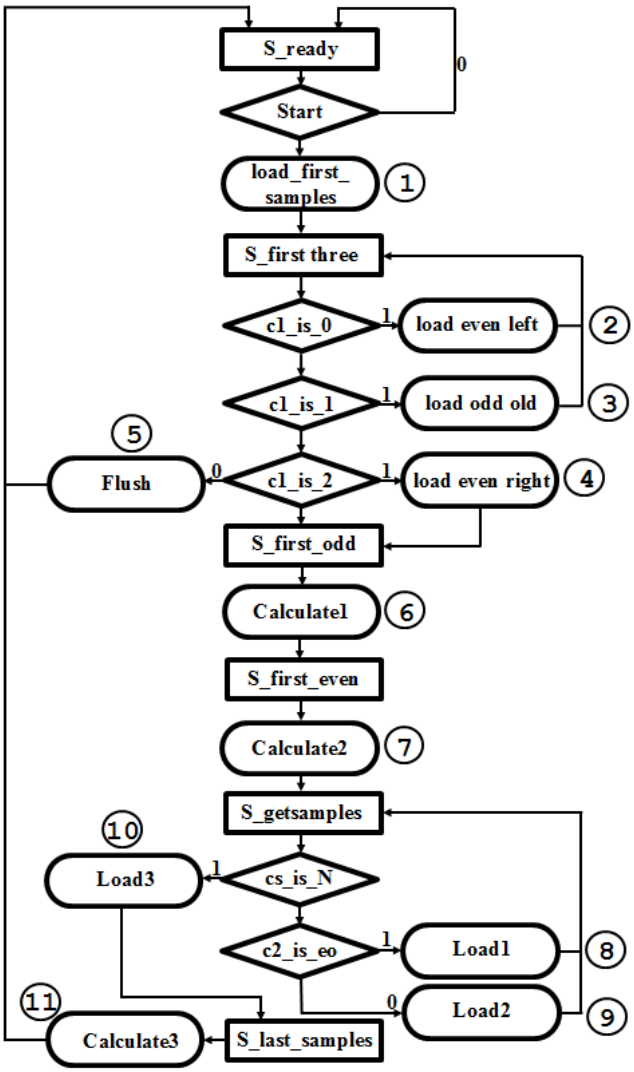

Figure 3. Algorithm State Machine Diagram of the architecture

At each clock cycle and by the help of the related control signal, next sample is taken and also one of the new sample values is calculated, and the register transfers are carried out. Index counters which are datapath controls are related to the memory unit and these provide to have been taken the corresponding sample point.

TABLE I. DECISION CONTROL SIGNALS GENERATED BY DATAPATH

Signal Name	Operation
c1	first there sample counter
c2_is_eo	even-odd counter, adjusts the even and odd samples sequentially
cs	sample counter, gets selected number of input samples

Some processing units which fulfill the transform are utilized in the designed structure. The model can be seen in the Fig. 2 for the designed processing unit. Each processing unit has its own unique id which indicates the corresponding unit number. The components which are used to constitute the system architecture are explained.

Datapath unit: This is one of the main blocks of the system including the hardware resources which the operations are managed. Datapath performs the DWT with the help of the control unit signals. This unit includes pipeline registers which are required for the pipeline technique. In the datapath, the shifting units are utilized instead of the multiplier hardware. The shifting units are simple, efficient and faster considering the operation cost. The logic circuit schema of the operational datapath unit which is comprised of the simple datapath components is shown in the Fig. 4.

TABLE II. DATAPATH OPERATIONS

Operation Number	Related Datapath Operation
1	c1<= 0 c2<= 0 csample <= 0
2	even_left <= datasamples c1<= c1 + 1 csample <= csample + 1
3	odd_old <= datasamples c1<= c1 + 1 csample <= csample + 1
4	even_right <= datasamples c1<= c1 + 1
5	c1<= 0 c2<= 0 csample <= 0
6	odd_new <= odd_old + ((even_left + even_right) >> 1) odd_reg <= datasamples c1<= 0 csample <= csample + 1
7	even_new <= even_left + ((odd_new + odd_new) >> 2) even_left <= even_right even_right <= datasamples odd_old <= odd_new csample <= csample + 1
8	odd_new <= odd_reg + ((even_left + even_right) >> 1) odd_temp <= datasamples c2<= c2 + 1 csample <= csample + 1
9	even_new <= even_left + ((odd_old + odd_new) >> 2) even_left <= even_right even_right <= datasamples odd_old <= odd_new c2 <= 0 csample <= csample + 1
10	odd_new <= odd_reg + ((even_left + even_left) >> 1) odd_reg <= datasamples c2<= c2 + 1 csample <= csample + 1
11	even_new <= even_left + ((odd_old + odd_new) >> 2) c2<= 0

Control unit: Control unit ensures that the scheduling and synchronization of all the datapath operations are accurate and also all the processes are executed without resulting any complexity. The control unit is a FSM (Finite State Machine) designed substantially. This FSM is designed taking into account the lifting based DWT which is fulfilled by the special purpose processor. The generated control signals are directed to the datapath and address unit to manage the transform. The control unit operates in a cooperative manner with datapath unit because this unit handles the events taken place in the datapath unit.

Memory unit: This unit stores the signal samples to be transformed. The address of the input sample which is under consideration is defined by the index value generated by the address unit. The new sample values are exchanged with the old values because of the transform is in-place and the old sample values are discarded. This part of the system has

been designed as an individual unit free from the datapath and control unit.

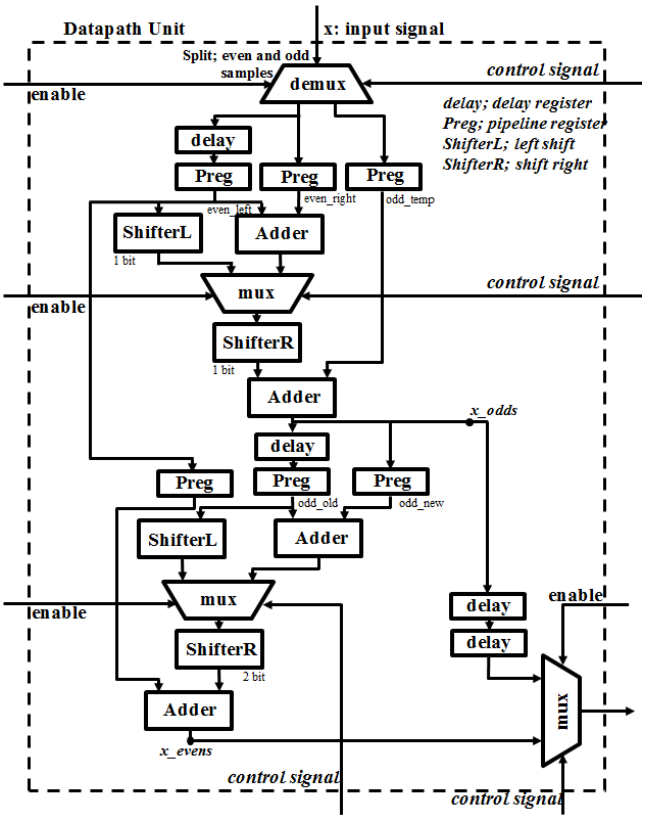

Figure 4. Datapath architecture of CDF 5/3 filter for symmetric boundary extension

The used address range depends on the input signal to be processed. Indeed, the memory can be assumed as a kind of LUT (Look Up Table) proceeding from bottom to top and a relevant illustration is given in the Fig. 5.

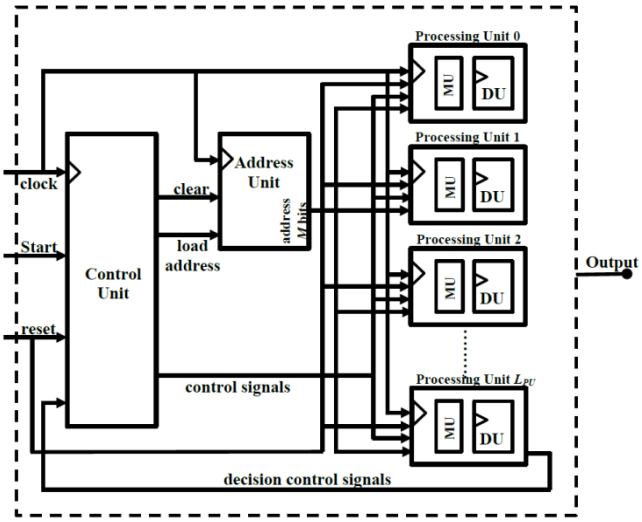

Figure 5. Model for the 2D CDF 5/3 filter architecture relating with processing unit s consisting with memory units (allocated memory address ranges) and datapath units

The results which are obtained from each processing unit are written to the same address ranges after processing the input samples. So, the memory requirements are reduced.

Address unit: Address unit has been added to the designed structure to provide that the progressing to 2D case from 1D is more easily. Address unit partitions the image into related number of image frames and transfers those frames to corresponding processing units to be processed in parallel. This unit is quite helpful for the interactions between the memory and datapath operations in case of multiprocessor system architecture. The address unit generates the addresses for the sample points which are taken from the memory unit. The related sample is taken from the generated address and directed to the datapath to be processed. This unit is also triggered with the clock signal and directed by the control signals from the control unit.

The address unit with a quite simple structure could be thought as a kind of index counter directed by the control signals, simply.

$$M = log_2 N \tag{8}$$

Where N is the number of sample points and M is the number of address bits. Basically, the registers are the data storage components and buffers.

The realization of the inverse transform is possible with some minor changes. It is enough only to change the shift amount values of the shifting elements in the datapath. So, the inverse transform could be performed by utilizing the same circuit.

B. Proposed 2D lifting based architecture

The 2D architecture is composed of almost the same components with the 1D architecture. Solely, 2D signals which are images are processed by virtue of a few additional control signals and multi-processing units to operate in parallel. The 2D signals are considered as matrix and the elements of matrix lay out from bottom to top in the memory.

To perform the lifting based DWT while working with images beforehand the wavelet transform coefficients of all the rows of the matrix are computed after that operation the wavelet transform coefficients of all the columns of the matrix are computed. Therefore, the approximation coefficients and horizontal, vertical and diagonal coefficients of image are computed.

The 2D architecture design has been composed of almost the same components with the 1D architecture design. 2D signals which are generally images has been processed by using a few additional control signals and multiprocessing units which operate in parallel. The 2D signals are considered as matrix, and the matrix elements placed from lower addresses to higher addresses, in the memory.

To perform the lifting based DWT while working with images or 2D signals, at first step the wavelet transform coefficients of all the rows of the matrix are computed. After that operation the wavelet transform coefficients of all the columns of the matrix are computed. Thus, the approximation, horizontal, vertical and diagonal coefficients have been computed for 2D signal input.

So, the datapath and control unit have to operate in cooperation, and also that control unit has been used for all the other processing units to coordinate other operations. Decision signals which are obtained from one of the processing units are sufficient to direct the main control unit because all the other processing units fulfils the same processes by operating in parallel and synchronously. As well, a single address unit provides carrying out all the datapath operations for the lifting based wavelet transform.

The samples stored at the memory unit (MU) are shared by each processing unit by the help of the address unit whereby the operations are executed in parallel. Each processing unit could reach only its permitted address range to prevent a problem. The number of the processing unit (PU) is given by L_{pu}. The rows with correct order and quantity have to be transferred to each processing unit. Because the number of the processing units, L_{pu} are less than the row number of the 2D signals, in general. These operations are accomplished by the help of the control and address units. A scalable structure has been formed by adjusting the proper number of PUs. Today's easily affordable multiprocessor structures are similar to these units. Consider an image which is the size of N_{row}xN_{column}, and the necessary memory unit which is the size of (N_{row}xN_{column})x1. Each individual image row is send to its dedicated PU so; the control passes to the next row after finishing the process of a previous row. PUs could fulfil their assigned task by reaching only their authorized MU ranges. Each PU has a right to reach only N_{row}/ L_{pu} number of rows.

The second step begins when all the rows are finished and sent to the in-place memory. All the columns are sent to the memory after the processing in the second step. The same hardware architecture is used for this second step by using the first step output coefficients as the input of the second step. The architecture is called folded structure when a single architectural layer is used for more than one transform step in this manner [27], [28].

The lowpass and highpass coefficients are obtained after the first step. The transpose of the first step output, wavelet coefficients, are applied to the input of the same circuit to complete the 2D transform. The desired output values after the transform are approximation (LL), horizontal (LH), vertical (HL) and diagonal (HH) coefficients revealing the information about the image.

Boundary issue: Another critical issue is the boundary extension method which affects the performance of the wavelet transform in case of 2D signal. The boundary extension problem corresponds to image samples which are placed at the edges. In this study, symmetric boundary extension method has been employed. The second sample and the (N-1)th sample have been used to calculate the transform coefficients at the frame boundaries, for the first and last samples. Actually, the relevant register content is used twice to overcome to this issue. The extension method is quite appropriate for the lifting schema. All the rows are processed in parallel and after finishing these steps the coefficient calculation of all the columns begins in order to avoid the control and computation overhead.

IV. Results And Discussions

Some experiments have been performed to test the performance of proposed system architecture. Designed architecture is synthesized for the Xilinx Spartan 3e FPGA board for this purpose. FPGAs offer high performance, flexible and balanced solutions compared to other common digital systems. The FPGAs have possibility of high level parallelism as well. Computation time is one of the crucial criteria to measure the system performance; it is defined as the total number of the clock cycles between the input and output time instants of the first sample.

1D signal data has been applied to the input of the designed architecture. The elapsed time which is the duration of having the first transformed sample after the first sample input takes 4 clock cycles. In general, the computation time is technology dependent. Total computation time is computed using $T=N\times T_{CLK}$ where N denotes the number of clock cycles, T_{CLK} is clock cycle time. The completion of the whole frame (for N point input signal) takes N+5 clock cycles beginning from the first sample input. The original form of the signal samples, the reconstructed signal with 256 points are shown in the Fig. 6. The superimposed form of the original and reconstructed signals is given also given in the Fig. 6. The obtained approximation coefficients and the detail coefficients are given in the Fig. 7. The approximation coefficients which are shown in the Fig. 7 are quite similar to the original signal and the detail coefficients reveal the fine details of the signal. The reconstruction error has not been obtained after the inverse transform.

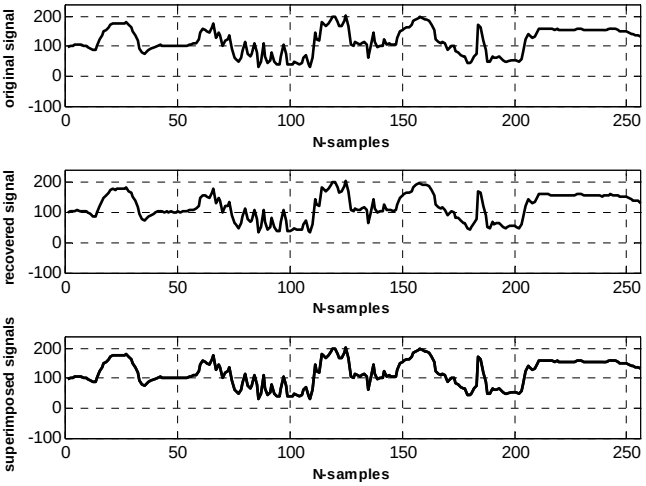

Figure 6. The original, the recovered and the superimposed signals to examine the forward transformation and the inverse transformation

The recovered signal sample values are assessed with an objective measure and RMSE (root mean square error) [40] is obtained as zero for the signal points and shown in Fig. 8.

Due to the characteristics of the CDF 5/3 filter this result has been obtained after recovering the image from lifting based wavelet transformation. A system which is designed to use for the lossless wavelet transformation and compression should provide that the recovered signal points are exactly the same as the original signal points, after inverse transformation. Here, this requirement is provided.

To change the shift amounts of the shifting units taking part in the datapath after reversing the sample point order of the transformed signal is quite effective. The shifting units are utilized to implement the multiplication and division operation when recovering the transformed signal to its original form. In the inverse transform, the sign of the scaling factors is changed, merge is exchanged with split, and the dataflow is reversed. The number of signal sample points, N, can be altered easily due to the proposed architecture which satisfies that system is scalable. The 2D architecture has been tested after the first step, and some images have been used for trials. Initially, the baboon image has been used for the 2D lifting based DWT.

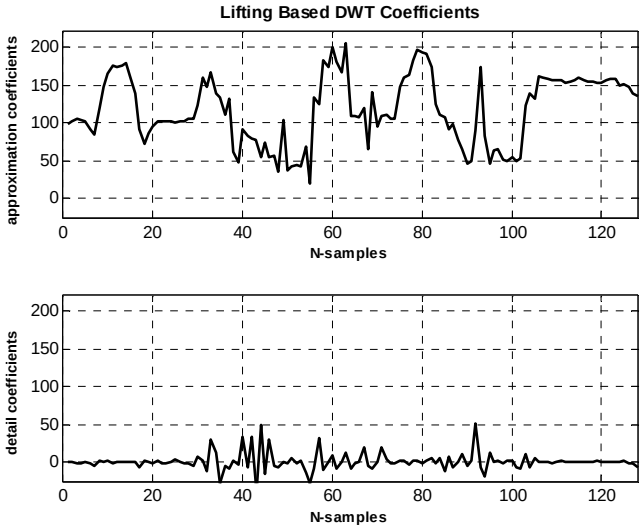

Figure 7. The approximation coefficients and the detail coefficients after transformation

Image which has 256×256 sample points defines the memory requirement at the same time. The number of PUs is chosen as four for the baboon image.

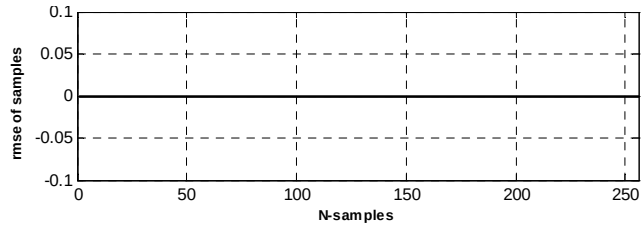

Figure 8. The RMSE error graphic of the original samples and the recovered wavelet coefficients

Some particular row ranges of image are assigned to each individual PU and the coordination of the operations are organized by means of the CU-AU pair. The load has been shared between each of the four PUs to reach the best performance. Predefined address ranges of memory are related to the corresponding PUs because the PUs can reach only the allowed memory space. The original image before wavelet transform is given in the Fig. 9. Approximation (LL), horizontal (LH), vertical (HL) and diagonal (HH) coefficients after transform are given in the Fig. 10.

In 2D (image) case, the total computation time for $N \times N$ signal is:

$$total\,comp.time = \left[\left(N + 5 \right) \times \left(\frac{N}{L_{PU}} \right) \right] \times T_{CLK} \qquad (9)$$

where L_{pu} denotes the number of processing units. The reconstructed image is shown in the Fig. 11.

When the obtained results are examined it is seen that the approximation coefficients are so similar to the original image. A horizontal, vertical and diagonal coefficient represents the related details of the image. After the first test the Lena image which is well known and frequently used in literature is chosen to test the constructed architecture. 512×512 sized grayscale image which is stored at the memory has been divided into equal sized rectangular frames. After that, equal number of frames has been transferred to each processing unit. The system component which has been designed as the address unit which adjusts the required range has been employed in order to prevent a breakdown of synchronization between memory unit and processing unit.

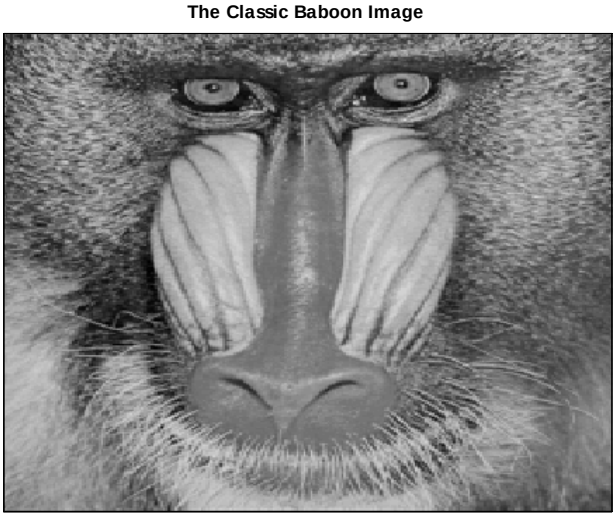

Figure 9. The original baboon image before transformation

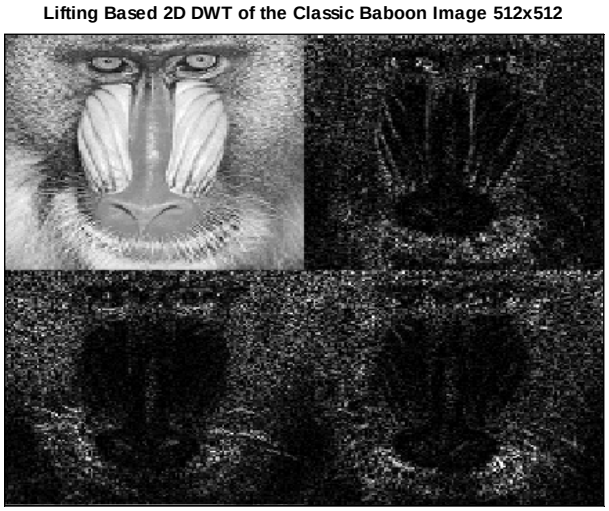

Figure 10. The approximation, horizontal, vertical and diagonal coefficients after the transformation

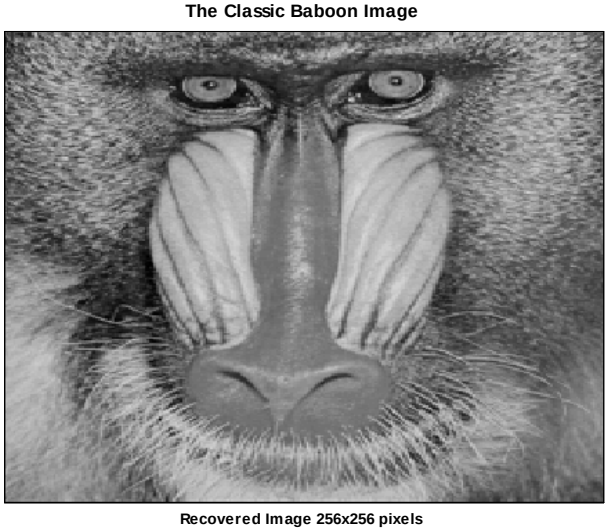

Figure 11. The original baboon image after the reconstruction (inverse transformation)

The number of PUs has been increased to eight using the scalability property of the architecture for the second image. The original image and the transform output are shown in

the Fig. 12 and Fig. 13, respectively. The reverse transformation is realized by virtue of the obtained coefficients and the resultant image after the reconstruction process is given in the Fig. 14.

Figure 12. The original Lena image before transformation

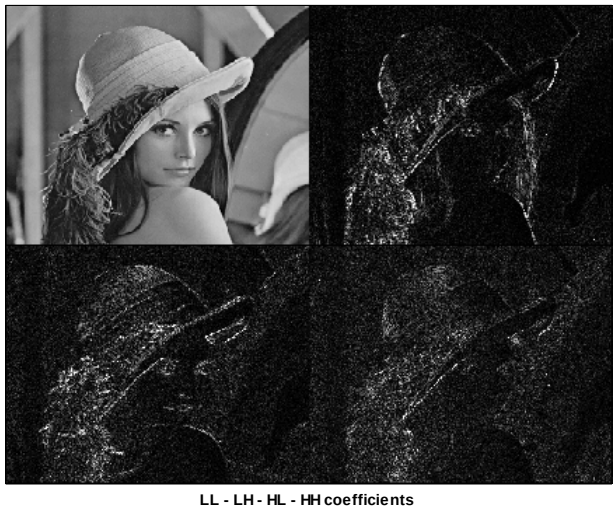

Figure 13. The approximation, horizontal, vertical and diagonal coefficients after the transformation

Figure 14. The original Lena image after the reconstruction (inverse transformation)

The original pixel values and the reconstructed image pixel values have been compared. For comparison purposes PSNR (Peak Signal to Noise Ratio) [40] is frequently used criterion in compression performance. RMSE graphic is supplied in Figure 15 rather than PSNR which goes to infinity for zero loss. As expected 2D RMSE is zero for all pixels.

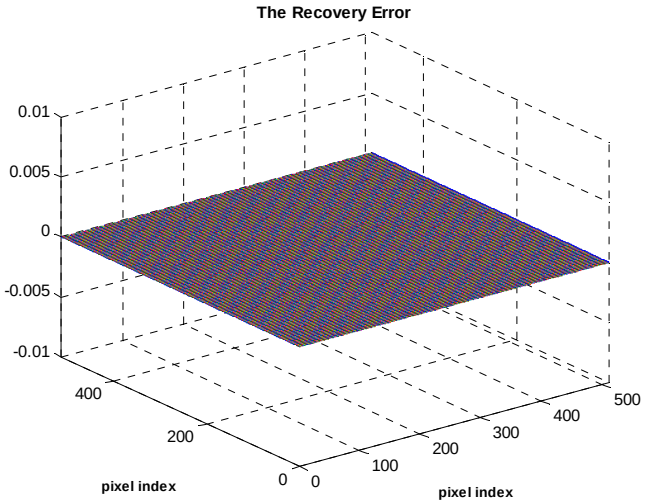

Figure 15. The error graphic for the reconstructed Lena image coefficients

V. Conclusion

In this study, an efficient multiprocessor pipelined system architecture for the lossless CDF 5/3 DWT filter is proposed. This structure has a simple datapath unit, and all the operations are shared onto designed processing units.

General system architecture has a hierarchical structure so that an effective scalability is provided. Thanks to scalability the number of PUs can be adapted where increasing number of PUs lead to decrease in computation time since they operate in parallel. Satisfying results have been obtained after conducting some test by using different size of images.

The usage of multiplier units has been avoided by using the characteristics of CDF 5/3 filter in favor of designed architecture when considering the system hardware. The control overhead and hardware complexity has been reduced by the simple construction of the datapath and control unit.

The symmetric boundary extension method has been used for the image edge points. So, the best system performance has been tried to achieve especially in 2D image case.

The same hardware components are used twice as folded architecture to use the hardware resources efficiently in 2D case. The same hardware resources have been used for the inverse transform but only changing the shift amounts of the shifting units. In addition, the pipeline technique has been utilized at the datapath either to avoid the time delays or to provide the efficient use of hardware resources.

As a consequence, the proposed architecture performs the operations of the widely used lossless CDF 5/3 filter with less hardware components, properly. It exhibits the better performance compared to the case without pipeline technique. The proposed architecture can be adapted efficiently for different applications which are based on wavelet analysis such as DWT based image compression, signal denoising, edge detection, speech recognition, biomedical image processing etc.

REFERENCES

[1] S.G. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation", IEEE Trans. Pattern Analysis Mach. Int., vol. 11, no.7, pp. 674-693, July 1989. doi: 10.1109/34.192463

[2] S.C.B. Lo, H. Li, M.T. Freedman, "Optimization of Wavelet Decomposition for Image Compression and Feature Preservation", IEEE Transactions on Medical Imaging, vol. 22, no. 9, September 2003. doi: 10.1109/TMI.2003.816953

[3] L. Cheng, D.L. Liang, Z.H. Zhang, "Popular Biorthogonal Wavelet Filters via a Lifting Scheme and its Application in Image Compression", IEE Proc.-Vis. Image Signal Process., vol. 150, no. 4, pp. 227-232, August 2003. doi: 10.1049/ip-vis:20030557

[4] T. Park, S. Jung, "High Speed Lattice Based VLSI Architecture of 2D Discrete Wavelet Transform for Real-Time Video Signal Processing", IEEE Transactions on Consumer Electronics, vol. 48, no. 4, pp. 1026-1032, November 2002. doi: 10.1109/TCE.2003.1196434

[5] T. Andre, M. Antonini, M. Barlaud, R.M. Gray,"Entropy-Based Distortion Measure and Bit Allocation for Wavelet Image Compression", IEEE Trans. on Image Processing, vol. 16, issue. 12, pp. 3058-3064, December 2007. doi: 10.1109/TIP. 2007. 909408

[6] G. Bhatnagar, Q.M.J. Wu, B. Raman, "A New Fractional Random Wavelet transform for Fingerprint Security", IEEE Trans. on Systems, Man, and Cybernetics-Part A: Systems and Humans, vol. 42, no. 1, pp. 262-275, January 2012. doi: 10.1109/TSMCA.2011. 2147307

[7] I. Ram, I. Cohen, M. Elad, "Facial Image Compression Using Patch-Ordering-Based Adaptive Wavelet Transform", IEEE Signal Processing Letters, vol. 21, no. 10, pp. 1270-1274, October 2014. doi: 10.1109/LSP.2014.2332276

[8] C.A. Garcia, A. Otero, X. Vila, D.G. Marquez, "A New Algorithm for Wavelet-Based Heart Rate Variability Analysis", Elsevier, Biomedical Signal Processing and Control, 8, pp. 542-550, 2013. doi.org/10.1016/j.bspc.2013.05.006

[9] E. Causevic, R.E. Morley, M.V. Wickerhauser, A.E. Jacquin, "Fast Wavelet Estimation of Weak Biosignals", IEEE Transactions on Biomedical Engineering, vol. 52, no. 6, pp. 1021-1032, June 2005. doi: 10.1109/TBME.2005.846722

[10] K.G. Oweiss, A. Mason, Y. Suhail, A.M. Kamboh, K.E. Thomson, "A scalable Wavelet Transform VLSI Architecture for Real-Time Signal Processing in High-Density Intra-Cortical Implants", IEEE Transactions on Circuits and Systems-I: Regular Papers, vol. 54, no. 6, pp. 1266-1278, June 2007. doi: 10.1109/TCSI.2007.897726

[11] J.A.T. Machado, A.C. Costa, M.D. Quelhas, "Wavelet Analysis of Human DNA", Elsevier Genomics 98, pp. 155-163, 2011. doi.org/10. 1016/j.ygeno.2011.05.010

[12] S. Saini, L. Dewan, "Application of Discrete Wavelet Transform for Analysis of Genomic Sequences of Mycobacterium Tuberculosis", Springerplus. 5: 64; 2016. doi: 10.1186/s40064-016-1668-9

[13] T. Meng, A.T. Soliman, M. Shyu, Y. Yang, S. Chen, S.S. Iyengar, J.S. Yordy, P. Iyengar, "Wavelet Analysis in Current Cancer Genome Research: A Survey", IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 10, no. 6, pp. 1442-1459, November/December 2013. doi: 10.1109/TCBB.2013.134

[14] I. Rodriguez, A. Manuel-Lazaro, A. Carlosena, A. Bermudez, J. Del Rio, S.S.Panahi, "Signal Processing in Ocean Bottom Seismographs for Refraction Seismology", IEEE Trans. Instr. Measurement, vol. 55, no. 2, pp. 652-658, April 2006. doi: 10.1109/TIM.2006.870107

[15] J. Ma, G. Plonka, H. Chauris, "A New Sparse Representation of Seismic Data Using Adaptive Easy-Path Wavelet Transform", IEEE Geoscience and Remote Sensing Letters, vol. 7, no. 3, pp. 540-544, October 2010. doi: 10.1109/LGRS.2010.2041185

[16] C.P. Uzunoglu, "Investigation of Degradative Signals on Outdoor Solid Insulators Using Continuous Wavelet Transform", Journal of Electrical Engineering and Technology, vol. 11, no.3, pp. 683-689, 2016. doi.org/10.5370/JEET.2016.11.3.683

[17] I.S. Uzun, A. Amira, "Framework for FPGA-Based Discrete Biorthogonal Wavelet Transforms Implementation", IEE Proc.-Vis. Im. Sig. Proc., vol. 153, no. 6, Dec. 2006. doi: 10.1049/ip-vis:20045080

[18] J.T. Olkkonen, H. Olkkonen, "Discrete Lattice Wavelet Transform", IEEE Transactions on Circuits ans Systems-II: Express Briefs, vol. 54, no. 1, pp. 71-75, January 2007. doi: 10.1109/TCSII.2006.883097

[19] H.I. Shahadi, R. Jidin, W.H. Way, Y.A. Abbas, "Efficient FPGA Architecture for Dual Mode Integer Haar Lifting Wavelet Transform Core", Journal of Applied Sciences 14 (5): pp. 436-444, 2014. doi: 10.3923/jas.2014.436.444

[20] K. Andra, C. Chakrabarti, T. Acharya, "A VLSI Architecture for Lifting-Based Forward and Inverse Wavelet Transform", IEEE Transactions on Signal Processing, vol. 50, no. 4, pp. 966-977, April 2002. doi: 10.1109/78.992147

[21] W. Sweldens, "The Lifting Scheme: A Custom-Design Construction of Biorthogonal Wavelets", Applied and Comp. Harmonic Analysis 3, no. 0015, pp. 186-200, 1996. doi.org/10.1006/acha.1996. 0015

[22] I. Daubechies, W. Sweldens, "Factoring Wavelet Transforms into Lifting Steps", The Journal of Fourier Analysis and Applications, vol. 4, issue 3, pp. 247-269, 1998. doi.org/10.1007/BFb0011095

[23] W. Zhang, Z. Jiang, Z. Gao, Y. Liu, "An Efficient VLSI Architecture for Lifting-Based Discrete Wavelet Transform", IEEE Transactions on Circuits and Systems-II: Express Briefs, vol. 59, no. 3, pp. 158-162, March 2012. doi: 10.1109/TCSII.2012.2184369

[24] K.A. Kotteri, S. Barua, A.E. Bell, E. Carletta, "A Comparison of Hardware Implementations of the Biorthogonal 9/7 DWT: Convolution Versus Lifting", IEEE Transactions on Circuits and Systems-II: Express Briefs, vol. 52, no. 5, pp. 256-260, May 2005. doi: 10.1109/TCSII.2005.843496

[25] O. Fatemi, S. Bolouki, "Pipeline, Memory-Efficient and Programmable Architecture for 2D Discrete Wavelet Transform Using Lifting Scheme", IEE Proc.-Circuits Devices Syst., vol. 152, no. 6, pp. 703-708, December 2005. doi: 10.1049/ip-cds:20059055

[26] J. Song, I.C. Park, "Pipelined Discrete Wavelet Transform Architecture Scanning Dual Lines", IEEE Transactions on Circuits and Systems-II: Express Briefs, vol. 56, no. 12, pp. 916-9, December 2009. doi: 10.1109/TCSII.2009.2035257

[27] B.K. Mohany, P.K. Meher, "Area-Delay-Power_Efficient architecture for Folded Two-Dimensional Discrete Wavelet Transform by Multiple Lifting Computation", IET Image Process, vol. 8, iss. 6, pp. 345-353, 2014. doi: 10.1049/iet-ipr.2012.0661

[28] G. Shi, W. Liu, L. Zhang, F. Li, "An Efficient Folded Architecture for Lifting-Based Discrete Wavelet Transform", IEEE Transactions on Circuits and Systems-II: Express Briefs, vol. 56, no. 4, pp. 290-294, April 2009. doi: 10.1109/TCSII.2009.2015393

[29] C.T. Huang, P.C. Tseng, L.G. Chen, "Flipping Structure: An Efficient VLSI Architecture for Lifting-Based Discrete Wavelet Transform", IEEE Transactions on Signal Processing", vol. 52, no. 4, pp. 1080-1089, April 2004. doi: 10.1109/TSP.2004.823509

[30] A. Darji, S. Agrawal, A. Oza, V. Sinha, A. Verma, S.N. Merchant, A.N. Chandorkar, "Dual-Scan Parallel Flipping Architecture for a Lifting-Based 2-D Discrete Wavelet Transform", IEEE Transactions on Circuits and Systems-II: Express Briefs, vol. 61, no. 6, pp. 433-437, June 2014. doi: 10.1109/TCSII.2014.2319975

[31] G. Dillen, B. Georis, J.D. Legat, O. Cantineau, "Combine Line-Based Architecture for the 5-3 and 9-7 Wavelet Transform of JPEG2000", IEEE Trans. on Circuits and Systems for Video Tech., vol. 13, no. 9, pp. 944-950, September 2003. doi: 10.1109/TCSVT. 2003.816518

[32] W.J. Laan, A.C. Jalba, J.B.T.M. Roerdink, "Accelerating Wavelet Lifting on Graphics Hardware Using CUDA", IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 1, pp. 132-146, January 2011. doi: 10.1109/TPDS.2010.143

[33] C.E. Kozyrakis, D.A. Patterson, "Scalable Vector Processors for Embedded Systems", IEEE Computer Society, vol. 23, issue no. 06, pp. 36-45, Nov./Dec. 2003. doi: 10.1109/MM.2003. 1261385

[34] K.C.B. Tan, T. Arslan, "Low Power Embedded Extension Algorithm for Lifting-Based Discrete Wavelet Transform in JPEG2000", IEEE Electronics Letters, vol. 37, Issue. 22, pp. 1328-1330, October 2001. doi: 10.1049/el:20010915

[35] W. Jiang, A. Ortega, "Lifting Factorization-Based Discrete Wavelet Transform Architecture Design", IEEE Trans. Circuits Syst. for Video Tech., vol. 11, no. 5, pp. 651-657, May 2001. doi: 10.1109/76.920194

[36] M. Dali, A. Guessoum, R.M. Gibson, A. Amira, N. Ramzan, "Efficient FPGA Implementation of High-Throughput Mixed Radix Multipath Delay Commutator FFT Processor for MIMO-OFDM", Advances in Electrical and Computer Engineering (AECE), vol. 17, no. 1, pp. 27-38, 2017. doi: 10.4316/AECE.2017.01005

[37] A.Y. Jean-Cuellar, L. Morales-Velazquez, R.J. Romero-Troncoso, R.A. Osornio-Rios, "FPGA-Based Embedded System Architecture for Micro-Genetic Algorithms Applied to Parameters Optimization in Motion Control", Advances in Electrical and Computer Eng. (AECE), vol. 15, no. 1, pp. 23-32, 2015. doi: 10.4316/AECE.2015. 01004

[38] A.R. Calderbank, I. Daubechies, W. Sweldens, B.L. Yeo, "Wavelet Transforms That Map Integers to Integers", Applied and Computational Harmonic Analysis 5, no. HA970238, pp. 332-369, 1998. doi.org/10.1006/acha.1997.0238

[39] W. Sweldens, "The Lifting Scheme: A Construction of Second Generation Wavelets", SIAM Journal on Math. Anal., vol. 29, issue 2, pp. 511-546, March 1998. doi: 10.1137/ S0036141095289051

[40] W.A. Pearlman, Wavelet Image Compression, Synthesis Lectures on Image, Video, and Multimedia Processing, Alan C. Bovik, Series Editor, Morgan & Claypool Publishers, 2013. doi: 10.2200/S00464 ED1V01Y201212IVM013