CHAPTER 1

INTRODUCTION

1.1 Overview

The growing of demand for bigger volume of data in different shapes like voice, image or, video requires a huge amount of storage and bandwidth for transmission. This leads to the need of a compression algorithm. Image compression algorithms play an important role in reducing storage by removing redundant data. One of the methods used in compression algorithm is Discrete wavelet transform (DWT) [1] [8] [9]. The wavelet transform decomposes signal into subbands with both time and frequency information. It supports various methods for analyzing the signal. Based on filter bank concept, wavelet transform can be implemented using the low-pass and high-pass filters. The characteristic of these filters depends on the wavelet shape.

DWT has been widely used in various fields such as in image compression, speech analysis, and pattern recognition because of its ability to decompose signal at multiple resolution levels. DWT is a multiple-level decomposition transform. In terms of implementation, it is done by repeated a process, execution a set of instructions. The implementation of DWT requires high cost of processing speed and energy consumption.

The type of DWT chosen depends on the signal used i.e 1-D DWT for the sound signal, 2-D DWT for an image, and so on. Also, 2-D DWT can be implemented using 1-D DWT by repeating it twice. There are many architectures proposed in literature that implement DWT. One of these approaches is the Filter design approach, which can be categorized into convolution-based, lifting-based,

and B-spline-based [10]. In convolution based, the DWT is implemented by filter directly. In the lifting-based method, it exploits the relationship between the low-pass and high-pass filters to save the number of multipliers and adders used. It has been repeated in [2], the one can reduce the multipliers is the B-spline factorization method.

The 2-D DWT can generally be realized by the separable [11] and non-separable [12] approaches. The separable approach splits the 2-D DWT implementation into two 1-D DWT operation (column-wise and row-wise) [13]. Memory requirement (for storing the intermediate data) and the critical part are essential for the 2-D or multi-dimensional transformation. The proposed algorithm in [7] utilized the interlaced read scan algorithm (IRSA) which changes the signal reading order from row-wise only into mixed row- and column-wise. Thus, this reduces the intermediate memory as a non-separable approach.

The Haar wavelet function is the simplest and the oldest orthonormal wavelet [14]. It appears very attractive in many applications as for example in medical application [15], image compression [16] [8], low energy image compression [17], the image coding, edge extraction, binary logic design [18], communication system [19], satellite-image fusion [20], de-noising the corrupted image by Speckle and Gaussian noise [21] and, removing the non-text pixels [22]. Based on transformed large image, the reduction in the resources usage of Haar 2-D DWT multilevel FPGA core can be used to counter severe hardware constraints of various wireless and mobile device applications [11]. Another kind of compression algorithm based on wavelet transform which divides the image to sub-image as $8x8$ matrix prior transformation. In these works [23],[24] ,and [25], the 2-D DWT is implemented on each sub-image.

1.2 Problem Statement

Processing large image requires a large amount of memory, power consumption and computationally intensive. Several architectures are proposed to achieve high speed with resource-efficient hardware. A filter is the main arithmetic component that has been used in status-of-the art DWT. One of the seeking points in DWT is how to reduce the number of arithmetic unit i.e the adder/subtractor, multiplier, and divider, as presented in [26]. These calculations reveal massive in energy consumption. Also, the chosen floating point or fixed point arithmetic operation effects the system performance as presented in [27]. Another aspect in discrete wavelet sub-bands to be considered is the calculation approximation coefficients [26]. For each DWT level, certain sub-bands will be generated, the same process is used in the next level. Unfortunately, this kind of approximation will effect the system performance but it has good impact on energy consumption. So, system will have a longer lifetime.

The way filter is designed affects the memory access and introduces longer delay. There are two types of memory in FPGA i.e the external memory and internal memory. The internal memory represents a big part of the hardware cost, whereas the external frame memory access consumes the most power [6]. Internal buffer stores intermediate data which is fully dependent on the filter design and DWT architectures as presented in [13] [6]. The methods used to access the image memory have high impact on energy consumption and speed of the transformation [7][13] and, internal buffer such as Row-column fashion with separable 2-D DWT [11] [6], RCCR [6], Line-based method [2] [6], Dual-model [7], Nonoverlapped and overlapped block-based scan [2] and, IRSA scan algorithm [7].

This thesis focuses on the hardware implementation for the Haar wavelet transform by using two different memory access method. The Haar wavelet is the simplest wavelet that leads to the reduction of energy consumption of the

system. This is suitable for application in the wireless sensors networks (WSN) [17], [23].

1.3 Research Objective

The main objects of this thesis are as follows:

i To implement a 2-D Haar wavelet transform for $64x64$ pixels image .

ii To reduce the number of image memory access by implementing the 2-D Haar wavelet transform with a suitable combination between using external memory and internal memory.

iii Targeting a low-power and high-speed architecture based on multi-levels Non-separable discrete Haar wavelet transform.

1.4 Scope of the Thesis

The thesis focuses on the hardware implementation of 2-D Forward / Inverse multi-levels discrete Haar wavelet transform with resource efficient using Verilog language. The final codes are tested for the Altera FPGA board with three decomposition levels. The design verified using 64x64, 128x128, 256x256, and 512x512 pixels image. Due to the overhead of off-chip memory (design time limitation), the design used the on-chip memory for image memory, it will be easy to handling memory addresses in order to get lower design time.

1.5 Thesis outline

This thesis is organized as the following; The fundamental wavelet concept, implementation techniques, and previous works are presented in Chapter 2. Chapter 3 describes the design methodology of the thesis. There are two architectures

proposed of accessing image memory i.e. the Line-based and dual-scan architectures. Chapter 4 presents the obtained results for both architecture and comparison is also made with the results from the previous works. Finally, chapter5 presents the work conclusion and the future work of the thesis.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Wavelet transform is a mathematical tool which used for signal analysis. It is used in several applications like image/video compression [1] and image denosing [28]. Image Compression is simply the way to minimize the storage or transmitted data to represent an image. Basically, there are two types of compression; Lossy and lossless. A lossless image compression will not able to reconstruct the original image perfectly after removing the redundant part of the image data. There are three type of redundant data; Spatial, spectral and, temporal redundancies.The decompression system is simply the inverse procedure.

The chapter is organized as the following; Discrete Wavelet Transform, Wavelet transform with the multi-resolution analysis and Image Pyramid, Wavelet based on filter banks, Haar wavelet transform. Finally, it discuses the previous work of hardware implementation of the Discrete wavelet transform.

2.2 Discrete Wavelet Transform

Wavelet transformation serves as an effective tool for Multi-resolution signal analysis which it used in application like data compression, image processing and information extraction as presented in [28] and [29]. The time-frequency character of the wavelet transform can be widely used to perform fine temporal analysis and fine spectrum analysis in high and low frequency. There are two main formulas to represent Wavelet Transform, Eq. 2.1 and Eq. 2.2 represent the Matrix formula and Figure 2.1 is shown the Filter banks representation.

$$Forward \quad Wavelet \quad Transform \quad W = \frac{1}{N} H * X * H' \tag{2.1}$$

$$Inverse \quad Wavelet \quad Transform \quad X = \frac{1}{N} H' * W * H \tag{2.2}$$

Where "$*$" represents matrix multiplication and "H'" is the transformation matrix transpose.

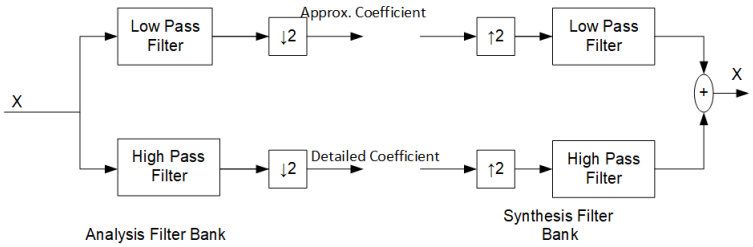

Figure 2.1: Level 1 wavelet transform by Filter Bank

Discrete Wavelet Transform (DWT) decomposes input signal into a number sub-bands referred to as low-pass sub-bands and high-pass sub-bands. The low-pass and high-pass sub-band components of a given DWT decomposition level are obtained by filtering the input signal using pair of low-pass filter(LPF) and high-pass filter(HPF). The low-pass filter and high-pass filter pair forms a quadrature mirror filter (QMF) structure to attain perfect signal reconstruction [1]. The shape of low-pass filter and high-pass filter and number of taps used to implement the filter, all depend on the wavelet shape.

The Hardware implementation of the wavelet transform mainly focuses on how to implement the filters with minimizing the number of multipliers/adders and minimal size of internal memory (data and temporal buffer). As shown in Figure 2.2, 2-D Filter bank block diagram is presented. Figure 2.3, Lena image is after 1^{st} decomposition level of Haar wavelet transform. As shown in Figure 2.3, the image divided after 1^{st} decomposition level into 4 sub-bands Low-Low sub-band (LL) as approximation component , Low-High band (LH) as vertical component, High-Low sub-band (HL) as horizontal component, and High-High

sub-band (HH) as diagonal component , LL sub-band locates at up-right, LH sub-band locates down-right, HL sub-band locates at up-lift, and HH sub-band locates at down-lift. As shown in Lena image after 1^{st} decomposition level, all diagonal, vertical, and diagonal components are represented the high-frequency components, so the image's edges appear with these components where edge represent abrupt transition (high frequency) in image .

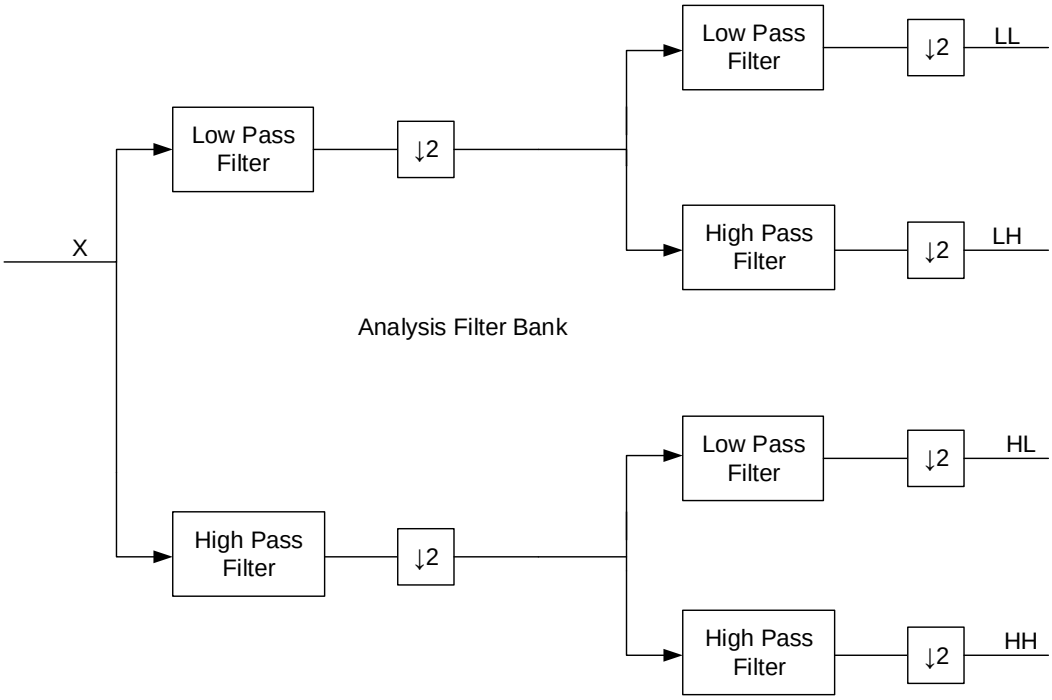

Figure 2.2: 1^{st} Level of 2-D wavelet transform by Filter Bank

Figure 2.3: Image after 1^{st}level of 2-D Haar Filter bank

2.2.1 Wavelet and Multi-resolution analysis

Multi-resolution signal analysis is how to represent the signal with different detailed as discused in [30], [1], and [31]. The signal can be represented with different detailed where the level order increases, the information or signal detailed decrease. Multi-resolution analysis can be used with different kinds of signal i.e 1-D signal like sound, 2-D signal like image, and 3-D signal like video. For image, the multi-resolution analysis is called Image pyramid.

Image pyramid is a simple structure to represent image in different resolution from full resolution NxN or 2^j x 2^j pixel image for j level (pyramid base) until 2^0 x 2^0 or one pixel to represent the image (the Top of pyramid) as Figure 2.4. The

following gray-scale Lena images Figure 2.5, Figure 2.6 and, Figure 2.7 are shown the same image with different resolution.

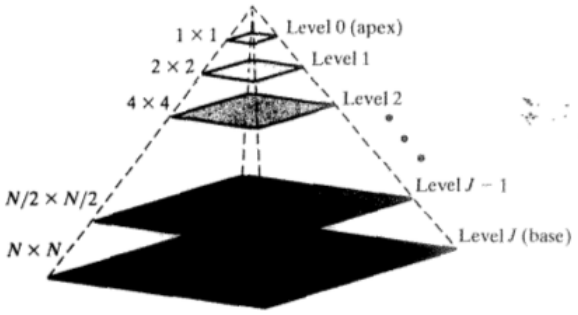

Figure 2.4: Image Pyramid [1]

Figure 2.5: High resolution gray-scale Lena image $256x256$ (Level j at pyramid)

Figure 2.6: Gray-scale Lena image with resolution $128x128$(Level j-1 at pyramid)

Figure 2.7: Gray-scale Lena image with resolution $64x64$(Level j-2 at pyramid)

2.2.2 Filter bank analysis based on 1-D DWT

Filters are one of the most widely used signal processing functions. Wavelets can be realized by iteration of filters with re-scaling. The resolution of the signal, which is a measure of the amount of detail information in the signal, is determined by the filtering operations, and the scale is determined by up-sampling and down-sampling (sub-sampling) operations. The 1-D DWT is applied on a signal, it decomposes that signal in two sets of coefficients; a low-frequency and a high-frequency set. The low frequency set is an approximation of the input signal at a coarser resolution, while the high-frequency set includes the details that will be used at a later stage during the reconstruction phase [3].

The multi-resolution DWT and inverse discrete wavelet transform (IDWT) can be consist of multiple levels of two channel filter bank, The analysis and synthesis filter bank are shown in Figure 2.8 where $H(z)$: low-pass filter and $G(z)$: high-pass filter, for perfect reconstruction [32] [2], The relationship between analysis and synthesis filter as following; respectively.

$$\tilde{H}(z) = G(-z) \tag{2.3}$$

$$\tilde{G}(z) = H(z) \tag{2.4}$$

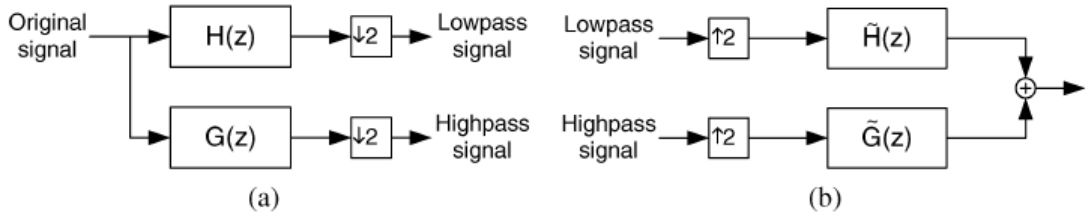

Figure 2.8: Two-channel filter bank for DWT and IDWT. (a) Analysis filter bank for forward DWT. (b) Synthesis filter bank for inverse DWT [2]

To explain the perfect reconstruction condition, Z-transformation is used to express the sub-band coding theory because Z-transform can easily handle changes

in sampling rate. According to the Z-transform and its sampling theorem, we can express the output of Synthesis filter bank (Inverse discrete wavelet transform) in terms of input of analysis filter bank (Forward discrete wavelet transform) as Eq. 2.5.

$$\tilde{X}(Z) = \frac{1}{2} * [H(z).X(z) + H(-z).X(-z)] * \tilde{H}(z) + \frac{1}{2} * [G(z).X(z) + G(-z).X(-z)] * \tilde{G}(z) \tag{2.5}$$

After arranging Eq. 2.5, Eq. 2.6 is produced, it is easy to extract the conditions for distortion free Eq. 2.7 and aliasing free Eq. 2.8.

$$\tilde{X}(Z) = \frac{1}{2} * [H(z).\tilde{H}(z) + G(z).\tilde{G}(z)] * X(z) + \frac{1}{2} * [H(-z).\tilde{H}(z) + G(-z).\tilde{G}(z)] * X(-z) \tag{2.6}$$

For Distortion free

$$H(z).\tilde{H}(z) + G(z).\tilde{G}(z) = 2 \tag{2.7}$$

For aliasing free

$$H(-z).\tilde{H}(z) + G(-z).\tilde{G}(z) = 0 \tag{2.8}$$

As a solution to Eq. 2.7 and Eq. 2.8, we got Eq. 2.3 and Eq. 2.4, so the output of synthesis part supposes to be equal to input signal of analysis part, it is called perfect reconstruction.

2.2.3 2-D DWT based on Filter bank

The 2-D DWT can be considered as a chain of successive levels of decomposition. Because the 2-D DWT can be represented as a separable transform, it can be computed by applying the 1-D DWT along the rows and columns of the input image of each level during the horizontal and vertical filtering stages. For

the case of three decomposition levels, the shaded areas in Figure2.9 represent the low-frequency coefficients that comprise the coarse image at the input of each level.

The following explains the steps of the decomposition level as shown in Figure 2.9. The input of level j is the low-frequency 2-D sub-band LL_j, which is the coarse image at the resolution of that level. In the first level, the image itself constitutes the LL image block (LL_0). The coefficients L(H), produced after the horizontal filtering at a given level, are vertically filtered to produce sub-bands LL and LH (HL and HH). The LL sub-band will either be the input of the horizontal filtering stage of the next level, if there is one, or will be stored, if the current level is also the last one. All LH, HL and HH sub-bands are stored, to contribute later in the reconstruction of the original image from the LL sub-band.

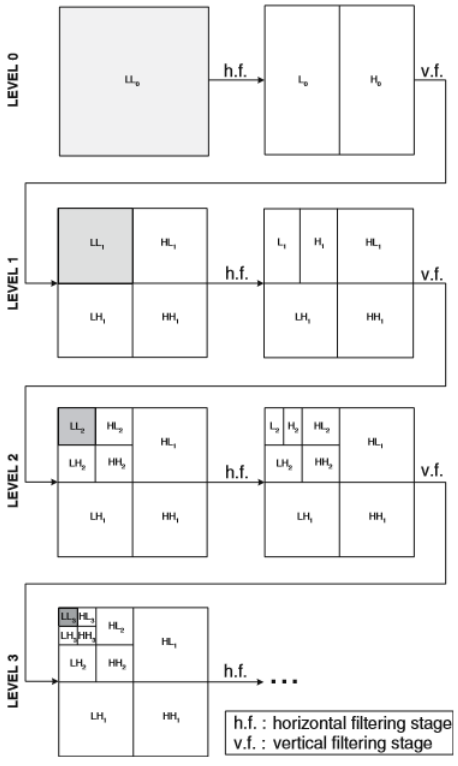

Figure 2.9: Diagrammatic representation of the decomposition for three decomposition levels [3].

2.3 Haar Wavelet Transform

Haar functions have been used from 1910 when they were introduced by the Hungarian mathematician Alfred Haar [18] [33]. The Haar transform is one of the earliest examples of what is known now as a orthonormal wavelet transform. The Haar function, being an odd rectangular pulse pair, is the simplest and oldest orthonormal wavelet.

Haar wavelet transform is used for medical application [15], image compression [16] [8], low energy image compression [17], de-noising the corrupted image by Speckle and Gaussian noise [21] and, removing the non-text pixels [22], it is used in communication system instead of IFFT/FFT [34]. JPEG is a compression algorithm, it combinates of DCT (Discrete Cosine Transform) and quantaization to produce a huge number of zeros in medium and high frequency region of transformed image. The proposed algorithm in [8], Haar wavelet matrix will be used instead of DCT.

Haar wavelet has two forms as other wavelet. First is the Matrix from as shown in Eq. 2.9 as explained in [1].

$$T = HFH \tag{2.9}$$

where :

— F : NxN image matrix.

— H : NxN transformation matrix.

— T : The resulting NxN transform / Coefficient matrix.

The transformation matrix H contains the Haar basis functions, $h_k(z)$. They are defined over the continuous, closed interval $z \in [0, 1]$ for $k = 0, 1, 2, \ldots, N-1$ where $N = 2^n$.

Converting continuous parameter z to discrete parameter k to generate H, where

— $k = 2^p + q - 1$

where

— $0 \leq p \leq n - 1$

— $q = 0$ or 1 for $p = 0$

— $1 \leq q \leq 2^p$ for $p \neq 0$

The Haar basis functions Eq. 2.10 Eq. 2.11

$$h_0(z) = \frac{1}{\sqrt{N}}, z \in [0, 1] \tag{2.10}$$

$$h_k(z) = \frac{1}{\sqrt{N}} * \begin{cases} 2^{p/2} & (q-1)/2^p \leq z < (q-0.5)/2^p \\ -2^{p/2} & (q-0.5)/2^p \leq z < q/2^p \\ 0 & \text{otherwise}, z \in [0, 1] \end{cases} \tag{2.11}$$

As a example 4x4 Haar transformation matrix, H_4, is

$$H_4 = \frac{1}{\sqrt{4}} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix} \tag{2.12}$$

In case of filter bank form, Haar wavelet defines as 2-taps low-pass filter Eq. 2.13 and high-pass filter Eq. 2.14.

$$h[n] = \frac{1}{\sqrt{2}}[1 \quad 1] \tag{2.13}$$

$$g[n] = \frac{1}{\sqrt{2}}[1 \quad -1] \tag{2.14}$$

2.4 Cyclone II Altera FPGA

Field Programmable Gate Array(FPGA)is used for testing or as a development prototype for any hardware architecture implementation.Once the FPGA hardware development stage passed, the ASIC (Application Specific Integrated Circuit) development stage follows. The FPGA gives design flexibility. It also provides testing capability and synthesis with different constraints. The FPGA is programmable the use of the Verilog language or any other hardware description language (HDL). One of FPGA available in the market for hardware architecture development is the Altera Cyclone II [35].

In this section, the overview of Cyclone II Altera FPGA structure is described. This overview will help to understand how is the FPGA handling the Verilog codes and the connections between memory and the arithmetic units. Also, it helps to realize the concept of limited resources especially for embedded multipliers since the number of the embedded multiplier is limited.

Figure 2.11 shows the top view of Altera Cyclone II DSP Development Board. In general, all Cyclone II devices contain a two-dimensional row and column based architecture to implement custom logic. The column and row interconnects of varying speeds provide signal interconnects between logic array blocks (LABs), embedded memory blocks, and embedded multipliers. Figure2.10 shows a diagram of the Cyclone II EP2C20 device. The logic array consists of LABs, with 16 logic elements (LEs) in each LAB. An LE is a small unit of logic providing efficient implementation of user logic functions. LABs are grouped into rows and columns across the device. The Cyclone II devices provide a global clock network and run up to four phase-locked loops (PLLs) [4].

The M4K memory blocks are the true dual-port memory blocks with 4K bits

of memory plus parity (4,608 bits). These blocks provide dedicated the true dual-port, simple dual-port, or single-port memory. Each embedded multiplier block can implement up to either two 9x9-bit multipliers, or one 18x18-bit multiplier. Each Cyclone II device I/O pin is fed by an I/O Element (IOE) located at the ends of LAB rows and columns around the periphery of the device. The number of M4K memory blocks,embedded multiplier blocks, PLLs, rows, and columns vary per device as shownin Table 2.1.

Table 2.1: Cyclone II FPGA resources

Feature / Device	LEs	M4k RAM blocks	Embedded multipliers	PLL
EP2C20	18752	52	26	4
EP2C70	68416	250	150	4

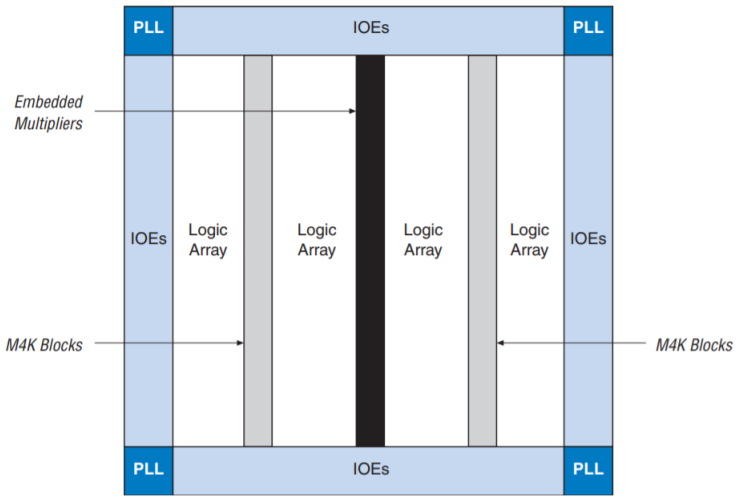

Figure 2.10: Cyclone II EP2C20 Device Block Diagram [4]

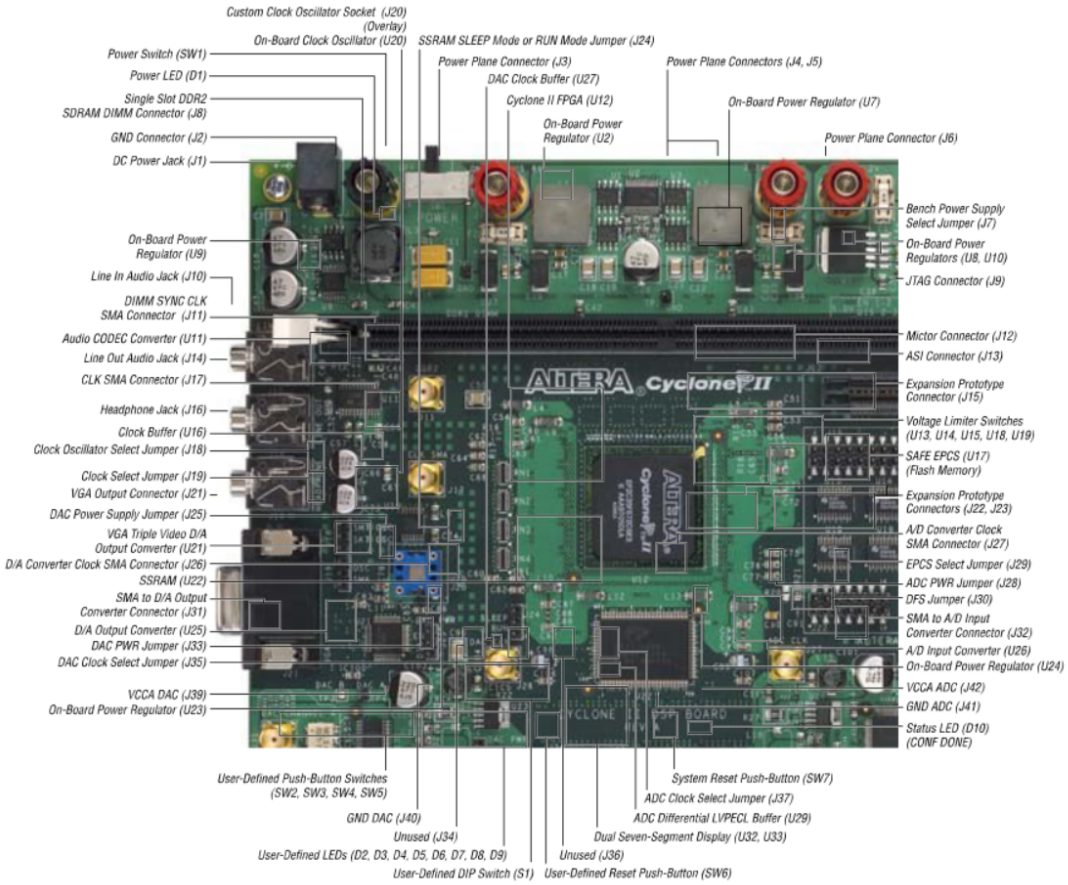

Figure 2.11: Cyclone II DSP Development Board [5]

The LE is the smallest unit logic in the Cyclone II architecture.It is compact and provides advanced features with efficient logic utilization. Figure 2.12 shows the content of the LE Cyclone II. The LE has two modes of operation i.e. the normal mode and arithmetic mode. The normal mode is suitable for general logic applications and combinational circuits. The arithmetic mode is ideal for implementing adders, counters, accumulators, and comparators [4].

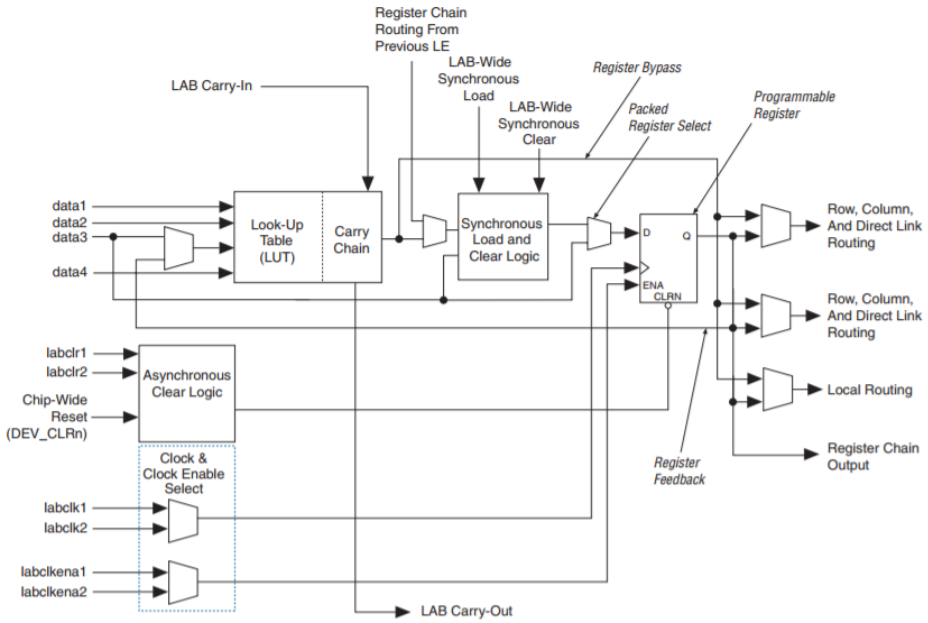

Figure 2.12: Cyclone II LE [4]

Figure 2.13 shows the structure of the logic array blocks (LABs). Each LAB consists of the following; 16 LEs, LAB control signals, LE carry chains, Register chains and, Local interconnect. The local interconnect transfers signals between the LEs in the same LAB. Register chain connections transfer the output of one LE's register to the adjacent LE's register within an LAB. The Quartus II Compiler places the associated logic within a LAB or adjacent LABs, allowing the use of local, and register chain connections for measuring performance and area efficiency [4].

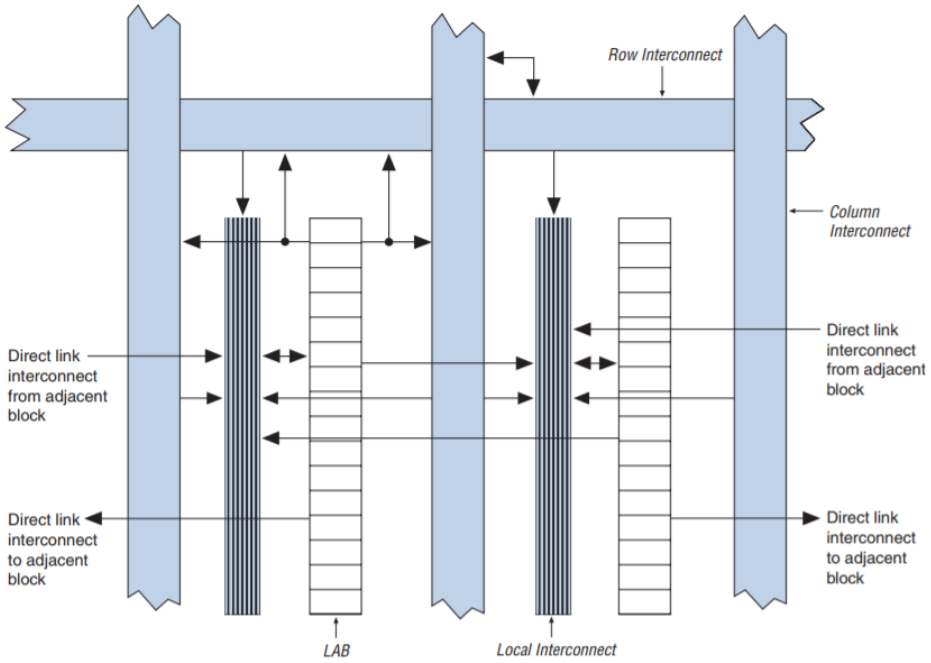

Figure 2.13: Cyclone II LAB Structure [4]

2.5 Review of Related Works

The Hardware implementation of the wavelet transform, in general, is dividing into to two approaches. The first approach is a filter design method, it mainly dedicates in fixed point representation of filter coefficients, reducing the number of DSP element used (Adder/subtractor and multiplier) by simplified the filter coefficients or getting a relationship between the low-pass filter and high-pass filter. The second approach is minimizing the internal buffer size and the number of image memory access, there is a relationship between achieving the second approach and filter design method. This work focuses on reducing the number of memory access times and internal buffer, the following section describes the different memory access ways.

2-D DWT hardware implementation, the memory issues including internal memory size and external memory frame access are the most critical problems. To get fast 2-D DWT, the reduction of number of image memory access times is required. In order to reduce the hardware cost, the internal memory size reduction is necessary. Mainly, the external memory is used for the image and internal

memory is used for buffering the intermediate results which are still in need.

2.5.1 Line-Based Scan Method

Line-based method scans the image line by line. The line could be column or row as presented in [2] [6] [13]. The simplest implementation of 2-D DWT is to directly perform 1-D DWT in the row direction and then in the column direction for each level of decomposition. In general, the image memory should be an off-chip memory. It has an issue in memory access because it will be had an impact on access time and power consumption. This issue leads to reduce the number of access time and force to use internal memory (data buffer and temporal buffer). To get better hardware utilization, there are different architecture to reduce the internal memory size. As explain in the following;

A Direct Architecture

The most straightforward method of 2-D DWT filter bank is to perform 1-D DWT in one direction and store the intermediate coefficients in the image memory, and then to perform 1-D DWT with these intermediate coefficients in the other direction to complete 1-level 2-D DWT [27] [6] [11]. For the next level decomposition,the low-pass low-pass (LL) sub-band is treated as the input signal, and the previous steps are performed recursively as illustrated in Figure 2.14. This architecture has the least hardware cost and no internal memory but it requires much external memory access. The bandwidth of external memory access including both read and writes, can be expressed as Eq. 2.15 where bandwidth (BW) is the number of reading/writing words (memory access) per image for all decomposition level [6].

$$BW = 4 * (1 + \frac{1}{4} + \frac{1}{16} + ... + (\frac{1}{4})^{(J-1)}) * N^2 \ (word/image) \tag{2.15}$$

where N is the image width and height of image, J is the number of decomposition levels.

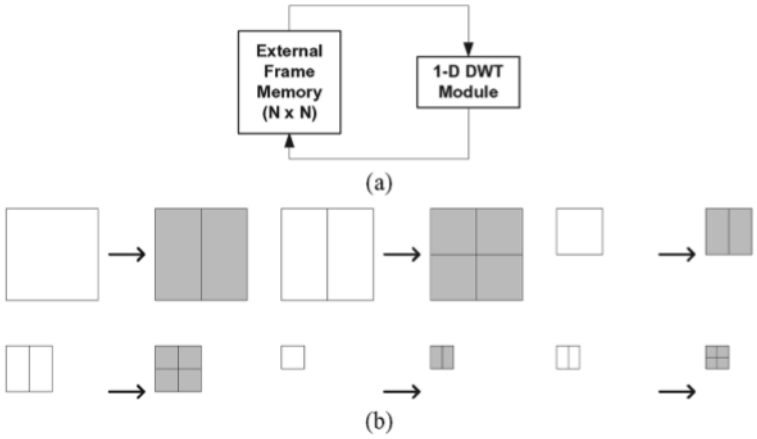

Figure 2.14: Direct 2-D implementation. (a) System architecture (the number in brackets represents the memory size in terms of words). (b) Data flow of external memory access (J =3; white and grey parts represent external frame memory reads and writes, respectively) [6]

The proposed architecture in [11] is a flexible hardware architecture of multi-level decomposition Discrete Wavelet Transform (DWT). In order to reduce computational complexities, the authors have chosen the Fast Haar Wavelet Transform (FHWT) in their scheme. In the 2D DWT implementation, in order to reduce resource usage a multilevel FPGA core can be used to counter severe hardware constraints.

Figure 2.15 shows the block diagrams of memory access technique, it is used in the proposed hardware implementation of forward DWT (FDWT) and inverse DWT (IDWT) based on the FHWT as presented in [11]. TheFDWT and IDWT hardware implementation utilized the row-column (RC) method. In this implementation, one-dimensional filter module is used on the rows first and followed by the columns (or vice versa). Figure 2.15 shows the image decomposition where the 1D DWT is applied on each row of the image pixel values. Then, the transformed data (coefficients) are taken column wise for another 1D DWT transformation process resulting of the first level of 2-D wavelet decomposition or four sub-bands.

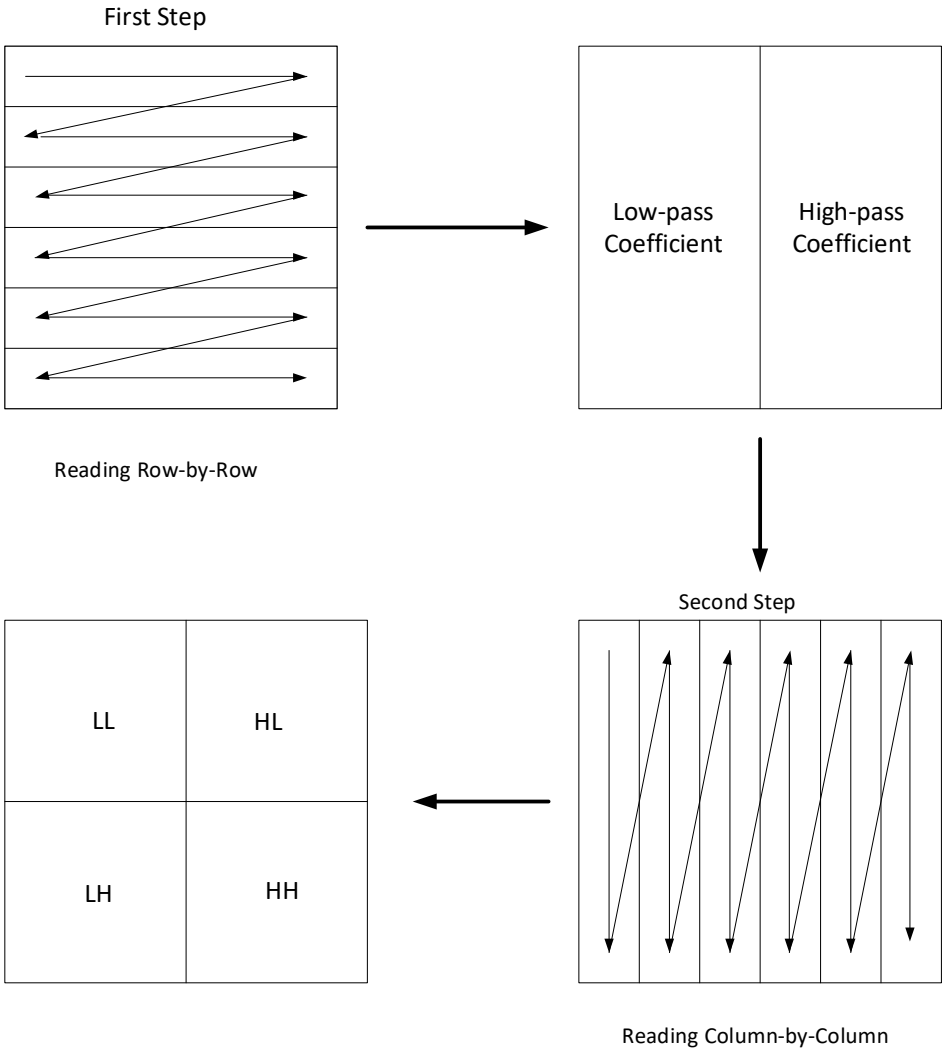

Figure 2.15: Row-Column (RC) Memory access fashion

The linear equations below describe the FDWT process;

$$L(i) = \frac{1}{2}(x(i) + x(i+1)) \tag{2.16}$$

$$H(i) = \frac{1}{2}(x(i) - x(i+1)) \tag{2.17}$$

and the linear equation below describe the IDWT process;

$$x(i) = L(i) + H(i) \tag{2.18}$$

$$x(i+1) = L(i) - H(i) \tag{2.19}$$

The FDWT/IDWT module consists of the adder, subtractor and right shifter that are used as the implementation to obtain the effect as the low-pass and high-pass filters.

B RCCR Architecture

The priority of row and column directions are identical. Thus, it is unnecessary to process the row coefficients first for every level decomposition all time, such as direct architecture. Instead, the priority can be arranged as row-column for the odd-level decomposition and column-row for the even-level decomposition as shown in Figure 2.16 [6].

The merging of two successive decomposition in the same direction can decrease the external memory access bandwidth by one half for every level, except the first level decomposition. The one of RCCR DWT implementation is to perform the former level decomposition and store the coefficients in a line buffer of size $N/2$, and then to perform the latter level decomposition with the stored coefficients. The external memory access bandwidth can be formulated by Eq. 2.20.

$$BW = (2 + 2 * (1 + \frac{1}{4} + \frac{1}{16} + ... + (\frac{1}{4})^{(J-1)})) * N^2 \ (word/image) \tag{2.20}$$