International Journal of Integrated Engineering, Vol. 4 No. 3 (2012) p. 26-34

Rapid Prototyping of Three-dimensional (3-D) Daubechies with Transpose-based Method for Medical Image Compression

Noor Huda Ja'afar[1,*], Afandi Ahmad[1], Abbes Amira[2,3]

[1]Department of Computer Engineering
Faculty of Electrical and Electronic Engineering
Universiti Tun Hussein Onn Malaysia (UTHM)
P. O. Box 101, 86400 Batu Pahat, Johor, MALAYSIA.

[2]Department of Electrical Engineering
College of Engineering, Qatar University
P. O. Box 2713, Doha, QATAR.
[3]NIBEC, University of Ulster, Jordanstown Campus, UK.

Received 1 October 2012; accepted 1 December 2012, available online 20 December 2012

Abstract: This paper presents an efficient architecture for three-dimensional (3-D) Daubechies with transpose-based method for medical image compression. Daubechies 4-tap (Daub4) and Daubechies 6-tap (Daub6) are selected with pipelined direct mapping design technique. Due to the separability property of the multi-dimensional Daubechies, the proposed architectures have been implemented using a cascade of three *N*-point one-dimensional (1-D) Daub4/Daub6 and two transpose memories for a 3-D volume of *N***N***N* suitable for real-time 3-D medical imaging applications. The architectures were synthesised using VHDL and implemented on Altera®Cyclone II (EP2C35F672C6) field programmable gate array (FPGA). An in depth evaluation in terms of area, power consumption, maximum frequency and latency are discussed in this paper.

Keywords: 3-D medical image compression, Daubechies, FPGA

1. Introduction

Efficient system architecture design for medical image compression has received a lot of attention [1]-[3] due to the more widespread use of three-dimensional (3-D) imaging modalities, such as magnetic resonance imaging (MRI), computed tomography (CT), positron emission tomography (PET), and ultrasound (US) that have generated a massive amount of volumetric data.

In these fields, both efficient storage and transmission of data through high-bandwidth digital communication lines are of crucial importance [2], [4]. Despite their advantages, most 3-D medical imaging algorithms are computationally intensive with matrix transformation as the most fundamental operation involved in the transform-based methods. Therefore, there is a real need for high-performance systems, whilst keeping architectures flexible to allow for quick upgradeability with real-time applications [5]. Moreover, in order to obtain efficient solutions for large medical volumes data, an efficient implementation of these operations is of significant importance.

Reconfigurable hardware, in the form of field programmable gate arrays (FPGAs) appears as viable system building block in the construction of high-performance systems at an economical price. Consequently, FPGAs seem an ideal candidate to harness and exploit their inherent advantages such as massive parallelism capabilities, multimillion gate counts, and special low-power packages [6], [7].

The aim of this paper is to develop an efficient reconfigurable architecture for Daubechies wavelet transform using pipelined direct mapping. An evaluation of these architectures in terms of area, power consumption, maximum frequency and latency are also carried out. Finally, this research is expected to propose a novel architecture of 3-D DWT using various wavelet filters and different design strategies that can be further applied as an intellectual property (IP) core for compression systems specifically in telemedicine applications.

The rest of the paper is organised as follows. An overview of the related work is given in Section 2. Section 3 explains the mathematical background for Daubechies. Section 4 exposes the proposed architecture of 3-D Daubechies. Experimental results and an analysis of the area, power consumption, maximum frequency as well as latency are presented in Section 5. Section 6 discusses the resulting outcomes. Finally, concluding remarks and further potential ideas to be explored are given in Section 7.

*Corresponding author: he100047@siswa.uthm.edu.my

2. Related Works

A close examination of the algorithms used in real-time medical image processing applications reveals that many of the fundamental actions involve matrix or vector operations [1]-[4]. Most of these operations are matrix transforms including fast Fourier transform (FFT), discrete wavelet transform (DWT) and some recently developed transforms such as finite Radon, curvelet and ridgelet transforms which are used in two dimensional (2-D) or three dimensional (3-D) medical imaging [8].

Unfortunately, computational complexity for the matrix transform algorithms is in the order from $O(N \times \log N)$ for FFT to $O(N_2 \times J)$ for the curvelet transform (where N is the transform size and J is the maximum transform resolution level) are computationally intensive for large size problems. For that reason, efficient implementations for these operations are of interest not only because matrix transforms are important in their own right, but because they automatically lead to efficient solutions to deal with massive medical volumes.

Since the aim of this research is on the implementation of 3-D medical image processing application using FPGA, a survey of the past, current and future works are concentrates on previously published FPGA implementation. In addition, the performances evaluations including area occupied, maximum frequency, power consumption and latency are considered during validate the proposed work with the existing works. Despite its complexity, there has recently been an interest in 3-D DWT implementation on various platforms. However, the previous and existing literature indicates that the works still in its infancy as algorithm development and software simulation.

An efficient architecture for 3-D Haar wavelet transform (HWT) using dynamic partial reconfigurable (DPR) is presented in [4]. HWT was selected to be implemented in this architecture because it is the simplest wavelet transform due to its simplistic algorithm. The proposed architecture was designed using pipelined direct mapping technique. Experimentally, the results indicated that using DPR mechanism the area and power consumption can be reduced although the number of input, N is larger. Interestingly, the proposed pipelined architecture has gives an overview of the operation system using Haar wavelet filter.

In [8], a wavelet-based compression scheme with an adaptive prediction (WCAP) is proposed. The proposed scheme applies a separable 3-D wavelet transform with high-pass and low-pass filter to a set of image that will produce a smooth signal and detailed signal respectively. Moreover, the CT, MRI and US are selected as image modalities with various pixel sizes and slice distances. This method consists of five stages which are correlation, lifting, predicting, quatising and adaptive arithmetic coding. Results obtained shows that the proposed method almost achieves the highest compression rates for CT, MRI and US. However, it is noted that the drawback of the scheme is the complex algorithm for compute the coefficients in each stage, thus will increase the area and memory required.

Another issue on the image compression is presented in [9]. A 3-D DWT approach is proposed for performing 3-D compression. A direct implementation of DWT is used for designing the first proposed architecture which consists of three set of filters. Each filter contains Multiply-Accumulate Cells (MACs) with latches for x-dimension and registers for y and z-dimension.

On the other side, the second proposed architecture is designed using a single pair of filters and it based on processing block data operation. Comparing both architectures performance, the control for first architecture is simple since the data are operated in a row-column operation but problems in memory requirement and latency are occurred. Meanwhile, the second architecture requires a small amount of storage depending on the filter size but the control is more complex compared to the first architecture. This paper gives an overview to design the proposed architectures for 3-D DWT.

As can be seen from the existing implementation [8]–[11], there still remains a huge gap for further research in exploiting reconfigurable computing for 3-D medical image compression and two major limitations can be identified as follows:

1. Medical image compression has not been intensively addressed in the existing 3-D DWT implementation.
2. Image compression is one of the well establish research area. However, medical image compression especially dealing with 3-D modalities is considered as a pre-mature research area.

3. Mathematical Background

The Daubechies wavelet transform is defined in essentially the same way as the Haar wavelet transform by computing running averages and differences via scalar product. The difference between them is the way that the scaling signals and wavelets are defined [12]. Interestingly, Daubechies wavelet transform has properties of longer supports for the scaling signals and wavelets.

One of the weaknesses of Daubechies wavelet transform is the edge problem that is discussed in the next subsection. Fortunately, the smaller number of wavelet tap can be used to avoid the edge problem. Thus, the Daubechies 4-tap (Daub4) and Daubechies 6-tap (Daub6) have been used in this study besides it is the most popular choice in medical imaging applications [12].

Daubechies 4-tap (Daub4) algorithm

The Daub4 wavelet is the simplest wavelet among the Daubechies wavelet families. Generally, Daub4 have four scaling signals and wavelets coefficients as given in equation (1) and (2) respectively.

$$h_0 = \frac{1+\sqrt{3}}{4\sqrt{2}}, h_1 = \frac{3+\sqrt{3}}{4\sqrt{2}}, h_2 = \frac{3-\sqrt{3}}{4\sqrt{2}}, h_3 = \frac{1-\sqrt{3}}{4\sqrt{2}} \quad (1)$$

$$g_0 = h_3, g_1 = -h_2, g_2 = h_1, g_3 = -h_0 \quad (2)$$

The 1-level Daub4 scaling signals and wavelets can be defined as follows:

$$\begin{pmatrix} h_0 & h_1 & h_2 & h_3 & 0 & 0 & 0 & 0 & & \\ g_0 & g_1 & g_2 & g_3 & 0 & 0 & 0 & 0 & & \\ 0 & 0 & h_0 & h_1 & h_2 & h_3 & 0 & 0 & & \\ 0 & 0 & g_0 & g_1 & g_2 & g_3 & 0 & 0 & & \\ 0 & 0 & 0 & 0 & h_0 & h_1 & h_2 & h_3 & & \\ 0 & 0 & 0 & 0 & g_0 & g_1 & g_2 & g_3 & & \\ 0 & 0 & 0 & 0 & 0 & 0 & h_0 & h_1 & h_2 & h_3 \\ 0 & 0 & 0 & 0 & 0 & 0 & g_0 & g_1 & g_2 & g_3 \end{pmatrix}$$

The scaling and wavelet functions are calculated by taking the inner product of the coefficients and data input values. In the last iteration, data input of s[N] and s[N+1] are not exist (they are beyond the end of the array) and cause the edge problem. To handle this edge problem, the data set is treated as it is periodic. The process of Daubechies wavelet using both scaling and wavelet functions are given as follows.

Step 1:

Consider the first row of a set of matrix:

$$f = \left(f(0), f(1), f(2), f(3), f(4), f(5), f(6), f(7)\right)$$
$$= \left(2,5,8,9,7,4,-1,1\right)$$

Step 2:

Extend the signal periodically:

$$\tilde{f} = \left(f(6), f(7), f(0), f(1), f(2), f(3), f(4), f(5), f(6), f(7)\right)$$
$$= \left(-1,1,2,5,8,9,7,4,-1,1\right)$$

Step 3:

Move the low and high pass filters along this vector, two steps at a time:

$$f_1 = \begin{pmatrix} h(0)f(6)+h(1)f(7)+h(2)f(0)+h(3)f(1) \\ h(0)f(0)+h(1)f(1)+h(2)f(2)+h(3)f(3) \\ h(0)f(2)+h(1)f(3)+h(2)f(4)+h(3)f(5) \\ h(0)f(4)+h(1)f(5)+h(2)f(6)+h(4)f(7) \\ g(0)f(6)+g(1)f(7)+g(2)f(0)+g(3)f(1) \\ g(0)f(0)+g(1)f(1)+g(2)f(2)+g(3)f(3) \\ g(0)f(2)+g(1)f(3)+g(2)f(4)+g(3)f(5) \\ g(0)f(4)+g(1)f(5)+g(2)f(6)+g(3)f(7) \end{pmatrix}$$
$$= \left(0.155,5.78,12.4,6.37,-0.837,0.966,0.871,-3.12\right)$$

Step 4:

Keep the last half of the vector f_1 fixed while low and high pass filter the first half of the vector (periodically extend the first half of the vector f_1):

$$\tilde{f_1} = \left(f_1(2), f_1(3), f_1(0), f_1(1), f_1(2), f_1(3), f_1(4), f_1(5), f_1(6), f_1(7)\right)$$
$$= \left(12.4,6.37,0.155,5.78,12.4,6.37,-0.837,0.966,0.871,-3.12\right)$$

Step 5:

Move the low and high pass filters along the first six elements of the vector, two steps at a time:

$$\begin{pmatrix} h(0)f_1(2)+h(1)f_1(3)+h(2)f_1(0)+h(3)f_1(1) \\ h(0)f_1(0)+h(1)f_1(1)+h(2)f_1(2)+h(3)f_1(3) \\ g(0)f_1(2)+g(1)f_1(3)+g(2)f_1(0)+g(3)f_1(1) \\ g(0)f_1(0)+g(1)f_1(1)+g(2)f_1(2)+g(3)f_1(3) \\ f_1(4) \\ f_1(5) \\ f_1(6) \\ f_1(7) \end{pmatrix}$$
$$= \left(10.6,6.87,-5.70,6.02,-0.837,0.966,0.871,-3.12\right)$$

Step 6:

Low and high pass filter the first quarter of the vector f_2 (periodically extend the first two elements):

$$\tilde{f_2} = \left(f_2(0), f_2(1), f_2(0), f_2(1), f_2(2), f_2(3), f_2(4), f_2(5), f_2(6), f_2(7)\right)$$
$$= \left(10.6,6.87,10.6,6.87,-5.70,6.02,-0.837,0.966,0.871,-3.12\right)$$

Step 7:

Low and high pass filter on the first four elements of the vector:

$$f_3 = \begin{pmatrix} h(0)f_2(0)+h(1)f_2(1)+h(2)f_2(0)+h(3)f_2(1) \\ g(0)f_2(0)+g(1)f_2(1)+g(2)f_2(0)+g(3)f_2(1) \\ f_2(2) \\ f_2(3) \\ f_2(4) \\ f_2(5) \\ f_2(6) \\ f_2(7) \end{pmatrix}$$
$$= \left(12.4,2.66,-5.70,6.02,-0.837,0.966,0.871,-3.12\right)$$

Daubechies 6-tap (Daub6) algorithm

The Daub6 wavelet is the most localised members among Daubechies wavelet families and it has six scaling signals and wavelets coefficients as given in equation (3) and (4) respectively, where $z_1 = \sqrt{10}$ and $z_2 = \sqrt{5+2\sqrt{10}}$.

$$h_0 = \frac{1+z_1+z_2}{16\sqrt{2}}, h_1 = \frac{5+z_1+3z_2}{16\sqrt{2}}, h_2 = \frac{10-2z_1+2z_2}{16\sqrt{2}}$$
$$h_3 = \frac{10-2z_1-2z_2}{16\sqrt{2}}, h_4 = \frac{5+z_1-3z_2}{16\sqrt{2}}, h_5 = \frac{1+z_1-z_2}{16\sqrt{2}} \quad (3)$$

$$g_0 = h_5, g_1 = -h_4, g_2 = h_3, g_3 = -h_2, g_4 = h_1, g_5 = -h_0 \quad (4)$$

The 1-level Daub6 scaling signals and wavelets are defined in the same way as Daub4 wavelet and the last iteration to compute the scaling signals is described in Equation 5. Since the scaling signals have length six, it would send h_2, h_3, h_4 and h_5 beyond the end of the array.

$$A^1_{N/2} = \left(h_2, h_3, h_4, h_5, 0, 0, ..., 0, h_0, h_1\right) \quad (5)$$

4. Proposed Systems Architecture and Implementation

System Overview Applications

Fig. 1(a) illustrates an application overview of the proposed medical image compression system including the transform, quantization and entropy coding blocks. In each block, buffers have been used for storing intermediate results to be processed. In the transform block, the input of 3-D images is transformed into wavelet coefficients. Then, the coefficients are quantised and finally coded to have the output bit stream. In this paper, the focus only concerns on the transform block and our goal is to propose an adaptive compression system for 3-D medical images, where all the blocks are reconfigurable.

System Architectures

The proposed system for 3-D Daub4 and Daub6 with transpose-based computation are illustrated in Fig. 1(b). The whole chain to calculate the 3-D Daub4/Daub6 gets an input as a 3-D image with $N \times N \times N$ point and outputs coefficients of the $N \times N \times N$ point. To simplify the hardware design, both the 3-D Daub4 and Daub6 are divided into three 1-D Daub4/Daub6 calculation cascaded together with transpose modules in between. This is achieved by performing the first 1-D Daub4/Daub6 along the rows (columns) of the array followed by 1-D Daub4/Daub6 along the columns (rows) of the transformed array.

The third 1-D Daub4/Daub6 are performed on corresponding pixels in each of the N sub-images that constitute the third dimension. All transpositions modules store the transposed coefficients into memory with a fetch unit module that reads back the coefficients for the next 1-D Daub4/Daub6 calculation.

Pipelined Direct Mapping Implementations

The 1-D Daub6 and Daub4 flow diagram with N-inputs sample for pipelined direct mapping implementation are depicted in Fig. 2 and Fig. 3 respectively. Both architectures include multipliers, shifters, registers and adders for their operation, with notation of 'Mul.', 'Shift.' and 'Add.' for multiplier, shifter and adder. The input to the 1-D Daub4/Daub6 is read row by row. Then, each of the input vector is processed to calculate the 1-D Daub4/Daub6 coefficients. The calculated 1-D Daub4/Daub6 coefficients are sent to the transpose module T_1 to perform matrix transpose. After that, the same process of 1-D Daub4/Daub6 is repeated by taking the output of the transpose module T_1 as their input. The output coefficients of the second 1-D Daub4/Daub6 are sent to the transpose module T_2. Finally, the third 1-D Daub4/Daub6 are performed on each N sub-images of the transpose 2-D coefficients.

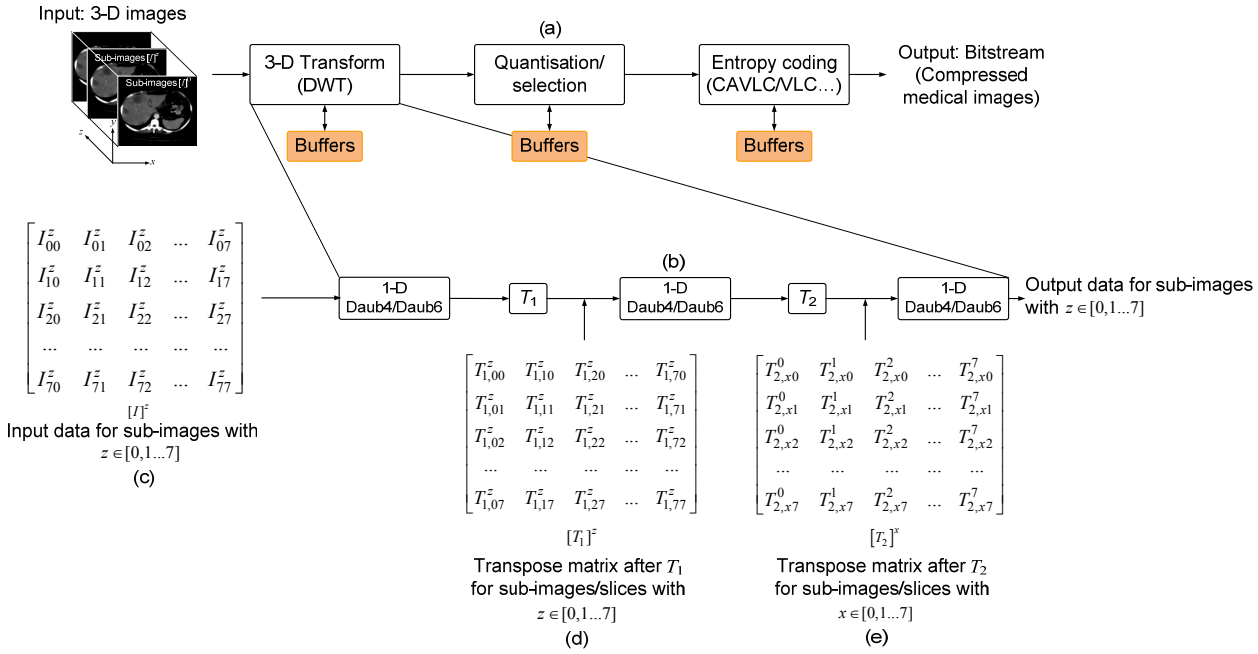


Fig. 1 Proposed system architectures
(a) Compression system overview (b) Architecture for 3-D Daub4/Daub6 with transpose-based computation (c) Input data for sub-images for $[I]_z$ (d) Transpose matrix after T_1 (e) Transpose matrix after T_2.

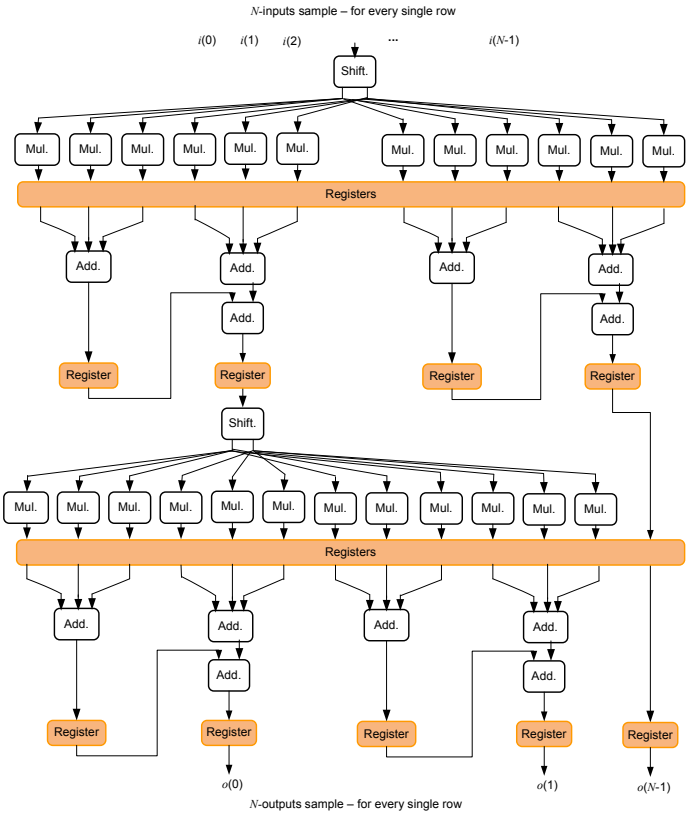


Fig. 2 Proposed system architecture 1-D Daub6 flow diagram with *N* inputs sample for direct mapped architecture.

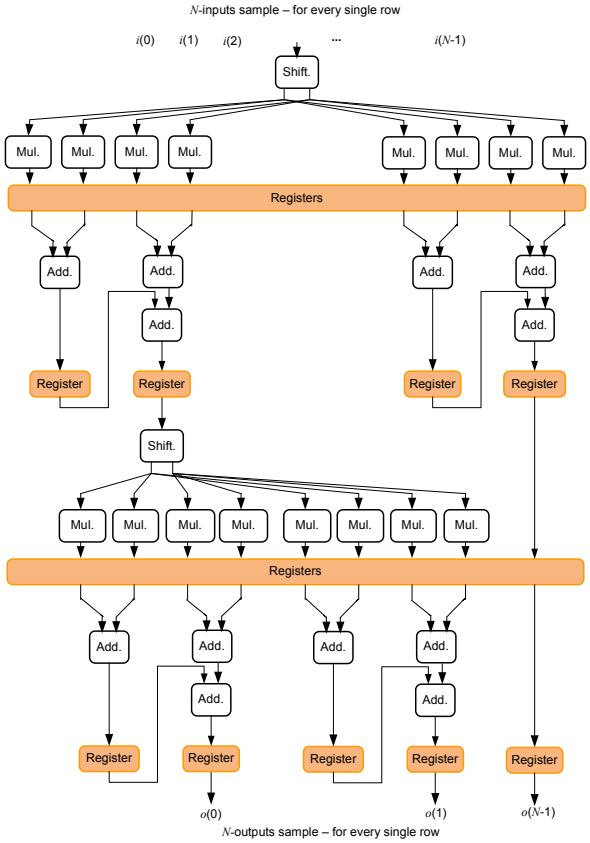


Fig. 3 Proposed system architecture 1-D Daub4 flow diagram with *N* inputs sample for direct mapped architecture.

5. Results and Analysis

Altera® Quartus II design flow has been used as a design flow reference and the proposed two architectures have been implemented on the Cyclone II (EP2C35F672C6). To evaluate the performance of the proposed architectures, four parameters have been selected including the area (LEs), maximum frequency (MHz), power consumption (mW) and latency (ns).

Table 1 lists the overall performance results for both proposed architectures for N = 4. As expected and considering their complex algorithm and edge problem, Daub6 utilise more LEs and total register to implement the proposed architecture. However, there is only 1% difference of area needed to implement the Daub4 architecture compared to Daub6 architecture. On the other side, a latency express the delivery time taken for a packet of data become the first available output data in the pipeline system. Referring to the waveform illustrated in Fig. 4(a) and (b), both Daub4 and Daub6 architectures necessitate 130ns to transmit the packet of data for produce an output. Whilst, Daub6 architecture save power consumption with 94.68mW and yield better maximum frequency at 30.17 MHz.

In summary, even though Daub6 required more area, it has higher vanishing moments that result in better signal approximation [13]. Thus, the proposed Daub6 architecture provides significant results in terms of power consumption and maximum frequency.

6. Discussions

To analyse visually the proposed architectures, both chip floor plan are given in Fig. 5(a) and (b). The chip planner floor plan uses a gradient colour scheme in which the colour becomes darker as the utilisation of a resource increases. Effectively, it can be clearly seen that the Daub4 implementation requires less complicated mapping in comparison with Daub6.

Concerning the higher vanishing moments of Daub6 [13], the implementation of 3-D Daub6 with transpose-based method on Altera®Cyclone II (EP2C35F672C6) FPGA yielding 0.63% better maximum frequency and consumes less power by 42.95% than Daub4 as illustrated in Fig. 6.

Altera® Quartus II PowerPlay Power Analysis tools are used for the purpose of power consumption estimation by read the verilog value change dump file (.vcd) and derives the toggle rate and static probability data. This .vcd file is created using ModelSim-Altera after the designs are synthesised and fitted to the target device.

Comparative study for both proposed architectures shows an imperative conclusion concerning the higher vanishing moment of Daub6. Analysis for the performance achieved in terms of area utilised, maximum frequency, power consumption and latency have reveals that with Daub6, complex designs can be implemented on FPGA and hence carry out a better performance achievements.

7. Conclusion

Two architectures for 3-D Daub4 and Daub6 have been proposed in this paper based on transpose computation for transform block of medical image compression. Comparative study for both architectures have reveals that Daub4 wavelet filter provides better achievements in terms of area than Daub6 wavelet filter, whilst in terms of power consumption, Daub6 wavelet filter consumes less power and directly yields better maximum frequency.

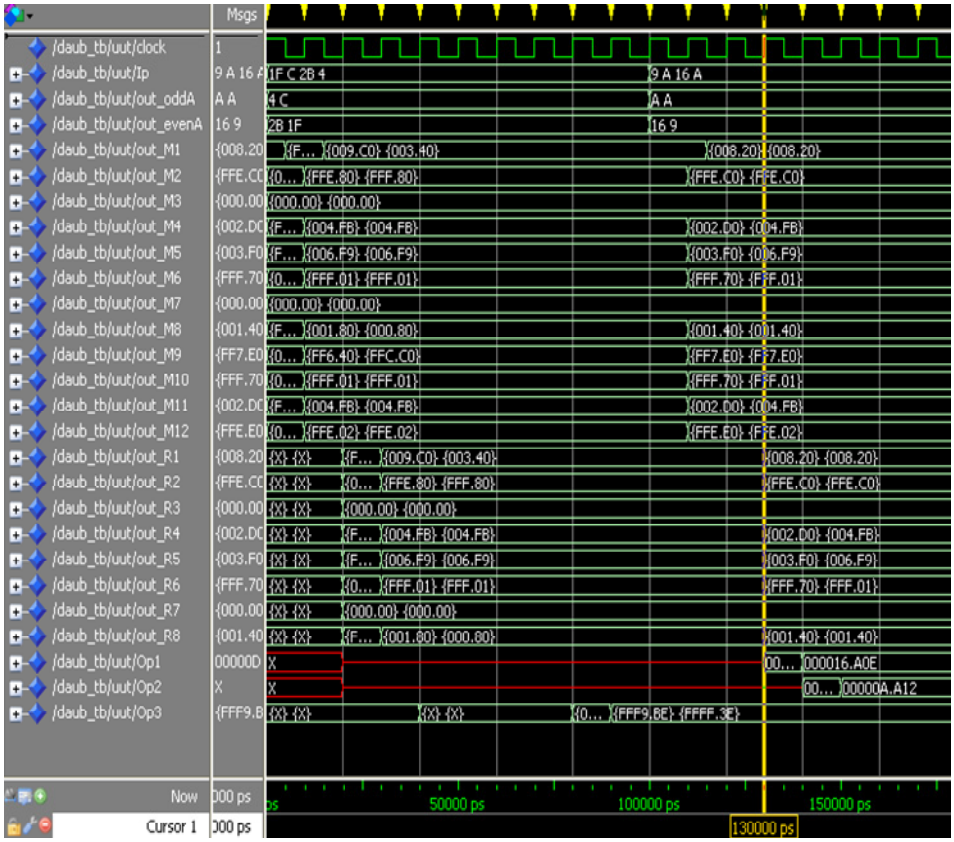
On-going research is focusing on the design and FPGA implementation of 3-D Daub4 and Daub6 using other arithmetic techniques such as distributed arithmetic (DA) and systolic design. Other wavelet filters such as Symlet, Coiflet and Biorthogonal as well as various transform size and real 3-D medical imaging modalities will be further explored to demonstrate the efficiency of the proposed architecture in medical imaging compression systems.

Acknowledgment

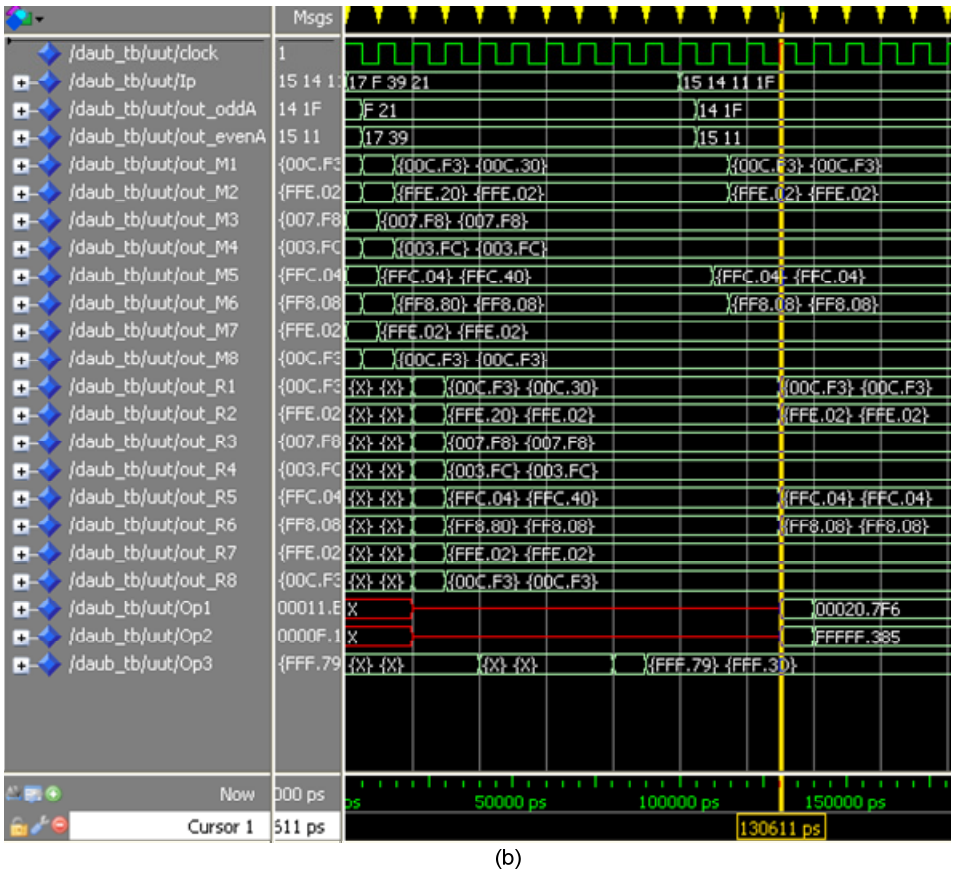
The authors would like to thank the Universiti Tun Hussein Onn Malaysia (UTHM) and Ministry of Higher Education Malaysia for funding this research work through Fundamental Research Grant Scheme (FRGS).

Table 1 Resources utilisation and overall proposed architectures performance on EP2C35F672C6 for N=4.

Parameters	Proposed 3-D architectures	
	Daub4	Daub6
Area (LEs)	1,978 (5%)	2,119 (6%)
Total registers	916	1051
Latency (ns)	130	130
Power consumption (mW)	165.97	94.68
Maximum frequency (MHz)	29.98	30.17



(a)



(b)

Fig. 4 Modelsim-Altera simulation for proposed architectures
(a) Daub4 (b) Daub6.

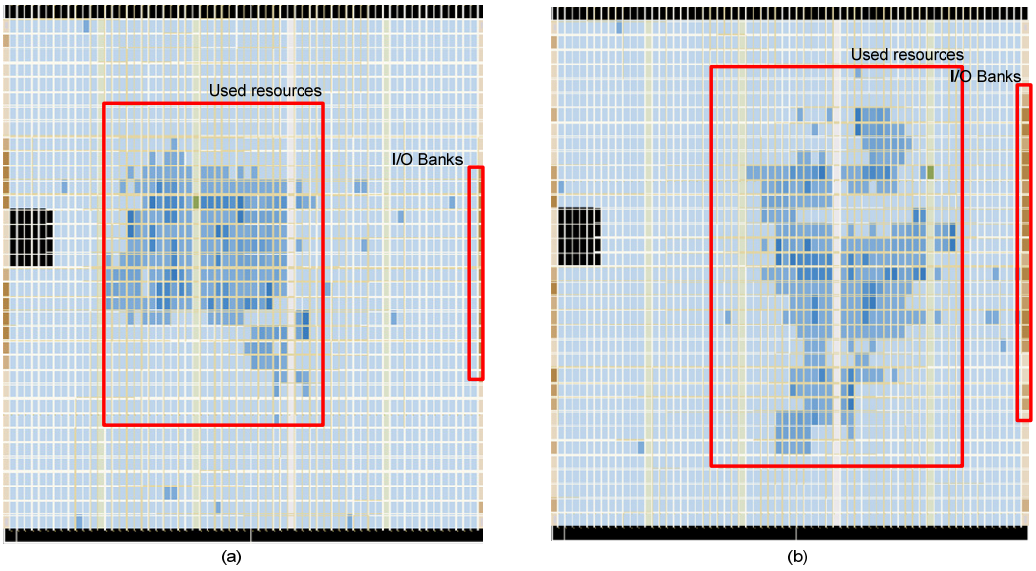


Fig. 5 Comparison of chip floor plan for $N = 4$
(a) Daub4 (b) Daub6.

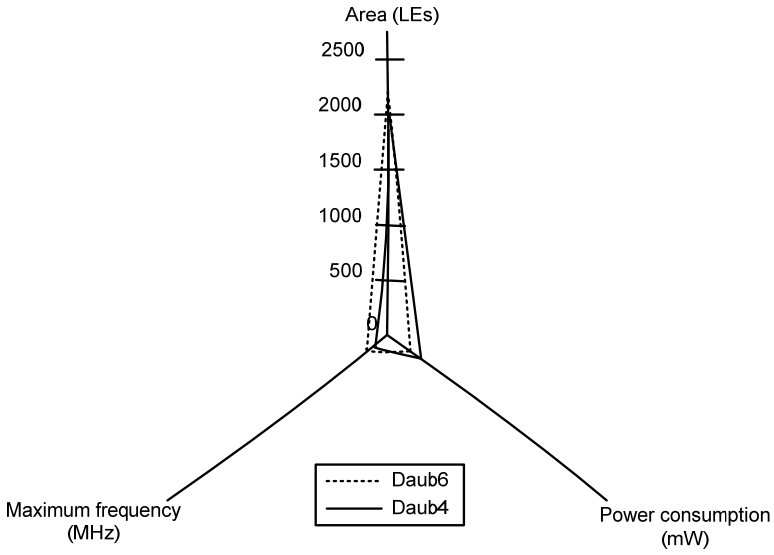


Fig. 6 Performance of area, maximum frequency and power consumption for $N = 4$.

References

[1] Sriraam, N., and Shyamsunder, R. 3-D medical image compression using 3-D wavelet coders. *Journal Digital Signal Processing*, Volume 21, (2011), pp. 100-109.

[2] Ahmad, A., and Amira, A. Efficient reconfigurable architectures for 3D medical image compression. *In Field-Programmable Technology, FPT 2009, International Conference*, (2009), pp. 472-474.

[3] Jiang, M., and Crookes, D. FPGA implementation of 3D discrete wavelet transform for real-time medical imaging. *In Circuit Theory and Design (ECCTD 2007), Proc. 18th European Conf. on,* (2007), pp. 519-522.

[4] Ahmad, A., Krill, B., Amira, A., and Rabah, H. Efficient architectures for 3D HWT using dynamic partial reconfiguration. *Journal of Systems Architecture*, Volume 56, (2010), pp. 305-316.

[5] Krill, B., Ahmad, A., Amira, A., and Rabah, H. An efficient FPGA-based dynamic partial reconfiguration design flow and environment for image and signal processing IP cores. *Journal of Signal Processing: Image Communication*, Volume 25, (2010), pp. 377-387.

[6] Dang, P. VLSI architecture for real-time image and video processing systems. *Journal of Real-Time Image Processing,* Volume 1, (2006), pp. 57-62.

[7] Todman, T., Constantinides, G., Wilton, S., Mencer, O., Luk, W., and Cheung, P., Reconfigurable computing: architectures and design methods. *Computers and Digital Techniques, IEEE Proceedings*, Volume 152, (2005), pp. 193-207.

[8] Chen, Y. T., and Tseng, D. C. Wavelet-based Medical Image Compression With Adaptive Prediction. *Computerized Medical Imaging and Graphics*, (2007), pp. 1-8.

[9] Weeks, M., and Bayoumi, M. Discrete wavelet transform: Architectures, design and performance issues. *The Journal of VLSI Sig. Process.*, Volume 35, (2003), pp. 155-178.

[10] Tian, D. Z., and Ha, M. H. Applications of wavelet transform in medical image processing. *In Machine Learning and Cybernetics, Proceedings of International Conference on,* (2004), pp. 1816-1821.

[11] Wang, J., and Huang, H. K. Three-dimensional medical image compression using a wavelet transform with parallel computing. *Proc. SPIE 2431 in Medical Imaging 1995: Image Display,* (1995), pp. 162-172.

[12] Jiang, M., and Crookes, D. Area-efficient high-speed 3D DWT processor architecture. *IEEE Electronics Letter*, Volume 43, (2007), pp. 502-503.

[13] Wahid, K. Low Complexity Implementation of Daubechies Wavelets for Medical Imaging Applications. *INTECH open*, (2011), pp. 122-134.

[14] Walker, J. S. A Primer on Wavelets and Their Scientific Applications Second Edition. (2008), pp. 54-58.

[15] Ismail, S., Salama, A., and ElYazee, M. A. FPGA implementation of an efficient 3D-WT temporal decomposition algorithm for video compression. *In Signal Processing and Information Technology, Proc. Conference IEEE International Symposium on,* (2007), pp. 154-159.