

Domain-Specific Reconfigurable Array Targeting Discrete Wavelet Transform for System-on-Chip Applications

Sajid Baloch^{1,2}, Imran Ahmed^{1,2}, Tughrul Arslan^{1,2}

¹School of Electronics and Engineering, University of Edinburgh, King's Buildings Mayfield Road, Edinburgh, EH9 3JL, UK

²Institute for System Level Integration, The Alba Campus, The Alba Centre, Livingston, Scotland, EH54 7EG, UK

Abstract

Domain-specific heterogeneous reconfigurable arrays are for one particular domain of applications, which provide high performance over generic Field Programmable Gate Arrays (FPGAs). This paper introduces an embedded reconfigurable array that targets discrete wavelet transform (DWT). Reconfigurable architectures are highly suitable for complex algorithms which are part of changing standards like JPEG2000 etc. The proposed reconfigurable array is flexible to implement lifting and integer based different DWT algorithms. Two benchmark DWT algorithms are implemented on three different platforms (on proposed array, on conventional FPGAs and on hardwired cores) for performance evaluation. The performance based on power consumption, timing and area, shows considerable improvement of presented arrays over FPGAs.

1. Introduction

Applications like Digital signal processing (DSP) and image processing are known as arithmetic-intensive activities. These computation intensive applications are facilitated by different technologies like application-specific integrated circuits (ASIC), general purpose DSP microprocessors, application specific standard products (ASSP), field programmable logic (FPL) devices which include field programmable gate arrays (FPGA) and general/application-specific reconfigurable SoC architectures. The demand for image/video applications in portable form has greatly increased in recent years. At the core of these productive and useful application is image/video compression technology. The DWT is one of these algorithms and technologies that had been developed for the compression of digital image/video data. Such features lead to significant interest in producing efficient algorithms for the realization of DWT hardware, for example, convolution based DWT, lifting based DWT and integer DWT etc. Each has its own merits and demerits and which make them suitable for different multimedia application. Current reconfigurable logic and Field Programmable Gate Array (FPGAs) provide quite powerful, flexible and cost-effective solutions for broad range of applications. The emerging trend in reconfigurable logic is Embedded Reconfigurable Arrays (RA). These RAs could benefit from parallel executions. Several companies [1][2] have introduced commercial embedded

FPGAs. Embedded Reconfigurable Array in System-on-Chip (SoC) can provide extra flexibility to the ASIC. This flexibility helps to accommodate post-fabrication modifications and also helps in reducing development time, debugging of errors, reduction in time-to-market and to add new functionality without going again into design stage. These all factors contribute in, reducing the overall cost of the product. Such flexibility is very useful for implementing complex algorithms which are part of changing standards like JPEG etc. However, Embedded FPGAs provide flexibility at the cost of power and area consumption, making them unsuitable for power critical applications like battery-powered devices and space related electronics. The introduction of RA for SoC platform has opened a new avenue to look for more domain-specific array structures. This is underlined by the observation that in most applications, the RA is used for one specific set of calculations that are known to the designer prior to the development of SoC. There are many applications, like hand-held devices and space related devices, where RA can be optimized for one domain of operation in such a way to reduce power and area consumption. This paper presents a domain-specific RA targeting Discrete Wavelet Transform (DWT). The proposed RA is flexible enough to accommodate Integer and lifting based DWT while maintaining high performance in terms of area and power consumption

2. Discrete Wavelet Transform (DWT)

Conventional DWT video compression is a 2-dimensional (2-D) process employed as a dyadic sub-band coding process. Conventional realization of DWT (Direct implementation) is also known as classical implementation. These processes are normally implemented as convolution based Finite Impulse Response (FIR) filters for both high and low band processing. The classical implementation of DWT is computationally intensive and area wasting as all the image pixels go through all of the 6 filtering processes [9] (as shown in Fig-1).

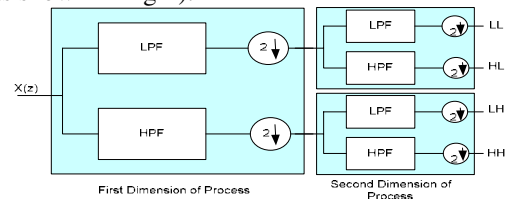


Fig-1 Classical (Convolution) two-dimensional DWT

However, most of the redundancy in the DWT system can be eliminated or minimized by Lifting based DWT [10].

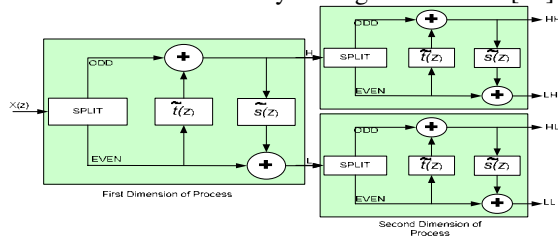


Fig-2 Lifting-Based two-dimensional DWT process

Both, classical and lifting based scheme has a common problem in terms of 9/7 filters that they require floating point operations. The floating point multiplication and other operations make it computationally complex. The Integer DWT algorithms provide the solution by changing the coefficients of filters in such a way that they become integer and hence avoiding the floating point operation [3].

3. Different Blocks of Reconfigurable Array

A simple DWT implementation on the proposed RA requires adders, subtractors, multipliers and dividers blocks. In order to accommodate a wide range of DWT algorithms, elements like ‘programmable buffers’ and ‘programmable normalizing blocks’ are also required. MATLAB simulations were performed to figure out the minimum bit width for lossless and lossy image processing for our architecture. It was decided on the basis of the simulation results that minimum 16-bits are required for 5/3 DWT and 20-bits for 9/7 integer based DWT (sign bit included) are required to conform with JPEG standard[11]. The proposed architecture provides programmable 24-bit operations for computation of different DWT algorithms. This provides more flexibility and precision for lossy image processing which can be an added advantage in critical applications like space and medical related image processing. The proposed array has three types of blocks/elements. These elements are described below;

3.1 Add-subtract Cluster

The add-subtract cluster can be configured as:

- parallel, digit-serial or bit-serial adder/subtractor
- can perform A-B and B-A operation

The basic module is 8-bits wide, three modules are grouped into cluster and configurable switches are provided between them to support cascading to get wider bit ranges (up to 24-bits). Even wider bit ranges are possible for different operations by cascading multiple clusters through mesh interconnect.

3.2 Coefficient Multiplier Cluster

Filter coefficients are multiplied through configurable coefficient multiplier clusters. Hardwired multipliers always give better performance in terms of power over the real multipliers. The cluster performs multiplication by carrying out a number of shift and addition operations. The floating point coefficients (for 9/7 lifting based DWT) can

be implemented through canonical-signed-digit (CSD) form [6]. The cluster has programmable shifter blocks, adder blocks and a multiplexer to accommodate a wide range of coefficients. The cluster can handle up to 24-bit operation to facilitate required precision. The cluster is shown below:

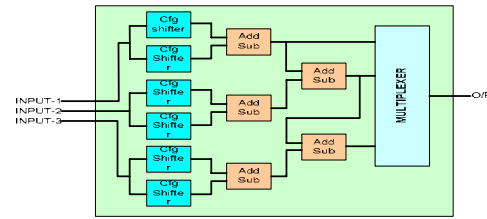


Fig-3 Coefficient Multiplier Cluster

Fig-3 shows the internal structure of the coefficient multiplier cluster. The cluster has three internal sub-module.

- Cfg-Shifter Module
- add-Sub Module
- Multiplexer Module

Cfg-Shifter performs the multiplication and division depending upon the algorithm. It can be configured to multiply or divide by any even integer value between 2 and 32. add-sub is the same as explained in section 3.1. The multiplexer is configured to select one of its inputs based upon the DWT algorithm. All internal modules are interconnected through programmable switches to incorporate different multiplications and divisions.

3.3 Configurable Buffer Cluster

The cluster can be programmed in 4-bit, 8-bit, 12-bit, 16-bit, 20-bit and 24-bit different combinations. A wider bit-range of operations can be handled by incorporating multiple clusters with the help of an interconnect mesh. The cluster can be used as buffer/delay element towards the hardware realization of a DWT algorithm. The cluster also provides configurable normalizing functionality depending upon the DWT filter type i-e 9/7.

4. Array for DWT

The uniformly arranged clusters in columns make it simpler to make manual routing and placement while configuring the array. The arrangement of the clusters in the array is done at the design-time and according to the application and required flexibility. The array is organized as shown in Fig-4. The elements of the array are interconnected through symmetrical configurable switches.

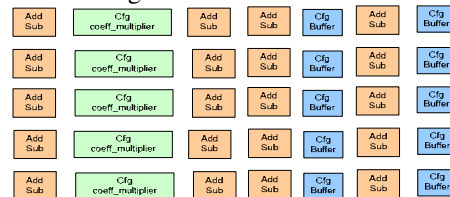


Fig-4 Arrangement of the clusters in the RA

Twenty four 4-bit tracks and twenty four 1-bit tracks are provided for both data and control lines. Connection boxes (C-Boxes) connect the pins of the cluster to the tracks, and the switch boxes (S-Boxes) connect together the intersection of the tracks [7]. The connection of clusters with tracks is illustrated in Fig-5.

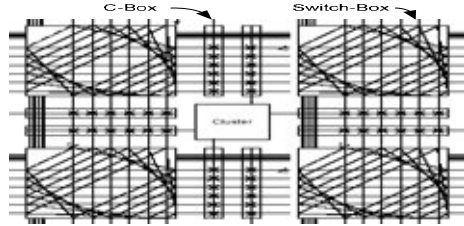


Fig-5 Basic mesh interconnect scheme

The C-boxes has flexibility $F_c=24$ and S-Boxes with $F_s=3$ [7]. Values selected for simplicity and sufficient flexibility to incorporate different algorithms of DWT. Different values can be selected by the designer depending upon the application requirements.

5. Implementation

5/3 Lifting Based DWT

5/3 lifting based DWT has many advantages over other DWT implementations such as, the 5/3 filter has short filter length for both low-pass and high-pass filter as compare to other JPEG2000 specified DWT filters i-e Daubechies 9/7 filter. 5/3 DWT based upon lifting based, are computed through following equations. $Y(2n)$ is even and $Y(2n+1)$ is odd term. Note that the JPEG2000's 5/3 even terms have an additional constant term of 0.5 [4].

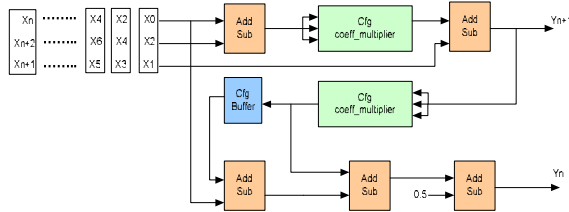


Fig-6 5/3 lifting based DWT implementation

The hardware realization in terms of our proposed array is shown in Fig-6 below.

The implementation is quite unique in its own as it requires less number of elements as compare to [5] and other architectures proposed so far.

9/7 Fast-IntegerDWT

Integer based DWT is an efficient approach which is based on computing power-of-two wavelet coefficients which are derived directly from the roots of the half-band filter [3]. The DWT is computed through following equation.

$$A = N * C * S * L * I^T$$

A denotes outputs of the two channel filter bank system and other definitions of different components of equation-1 are:

$$A = \begin{bmatrix} a_0(n) \\ a_1(n) \end{bmatrix} \quad C = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad S = \begin{bmatrix} s_4 \\ s_3 \\ s_2 \\ s_1 \\ s_0 \end{bmatrix} = \begin{bmatrix} 0 & -2^2 & 0 & 2^2 & 2^4 \\ 0 & 0 & 0 & 2^6 & 2^7 \\ 2^3 & 0 & -2^4 & 0 & 2^3 \\ -1 & 0 & -2^2 & 0 & 2 \\ 0 & -2^3 & -2^3 & 2^3 & 0 \end{bmatrix}$$

$$N = \begin{bmatrix} 2^{-8} & 0 \\ 0 & 2^{-7} \end{bmatrix} \quad I = [x(n-4) \dots x(n-1) \ x(n) \dots x(n+4)]$$

The array proposed in this paper is flexible enough to support number of different possible DWT implementations, such as 5/3 convolution and lifting based, 9/7 lifting based and 9/7 integer based DWT.

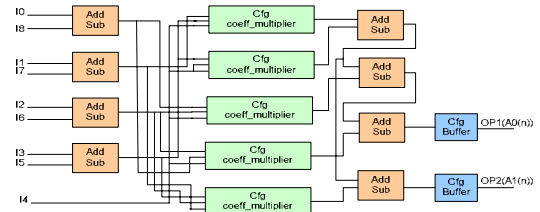


Fig-7 9/7 Integer DWT implementation on proposed RA

6. Performance Evaluation

Hardware realizations of two different DWT architectures were carried out on the proposed array to measure its performance. The same algorithms were implemented on different architectures (ASIC and Standard FPGA) and were compared.

6.1 Comparison with other systems

The two DWT algorithms are implemented on hardwired ASIC, commercial Xilinx Virtex-E FPGA [8] and on our proposed RAs. The performance in terms of overall power consumption and maximum operating frequency is shown in table-1 and table-2. All these systems use 0.18μm CMOS technology and run at 1.8v. The values are measured for single frame of the Lenna image 128x128.

	0.18μmASIC	Proposed RA	Xilinx Virtex-E
Area (μm ²)	121583	584321	627166
Power Consumption (mW)	2.38	6.13	9.98
Max Frequency MHz	272	123	98

Table-1 5/3 Lifting based DWT Performance Evaluation

	0.18μmASIC	Proposed RA	Xilinx Virtex-E
Area (μm ²)	151150	823788	877120
Power Consumption (Mw)	3.343	8.6	12.48
Max Frequency MHz	220	123	73

Table-2 9/7 Integer DWT Performance Evaluation

6.2 Power Consumption

The power consumption values measured for our array and for the hardwired implementation are obtained with post-routing simulation with typical switching activity and accurate parasitic and load information, using Cadence

Silicon Ensemble and *Synopsys PrimePower*. In the case of the Virtex-E FPGA, the design software by Xilinx is used to map the implementations. The values provided include the power consumed by the configuration circuit and configuration memory. Our proposed array consumes between 31% and 38% less power than the Virtex-E. This is mainly caused by the fact that less interconnects and larger clusters are used in our array. Nevertheless, the consumption of the reconfigurable implementations remains significantly higher than hardwired ASIC. This is caused by the configurable switches overhead.

6.3 Timing Analysis

Our array has a maximum frequency 21% to 40% higher than the Virtex-E, however it is still 45% to 56% less than the maximum frequency achievable in ASIC. The reduced frequency is understandable as it is because of the delays introduced due to reconfigurable switches.

6.4 Power and Area Distribution Among Clusters

The power and area measurement of different clusters were performed. The Coefficient multiplier cluster uses more area and power than buffer and add-subtract cluster as illustrated in Fig-8. This is expected as coefficient multiplier incorporated many shift and addition and subtraction processes.

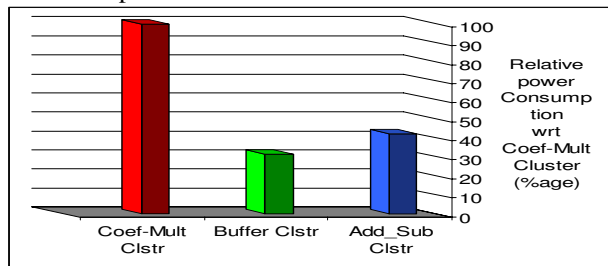


Fig-8 Average % Relative power consumption of different clusters with respect to Coefficient multiplier cluster

The Coefficient multiplier cluster used 70% more power than buffer cluster and approximately 58% more power than add-subtract cluster. When modules and clusters are un-configured and if they have no activity at their inputs, they exhibit only static power consumption and no dynamic one. In the case of un-configured C-boxes, some switching power is dissipated when the output of the cluster connected to the C-box is switching.

Fig-9 shows the area overhead used to make the hardware reconfigurable. The add-subtract cluster occupies only 6% of the total area while the C- and S-boxes occupy 50% and 44% respectively. As can be seen from the graph these area values include the area occupied by the configuration registers, which represents a large percentage of the area of the boxes. The total area can be reduced considerably if the flexibility of the C- and S-boxes is reduced.

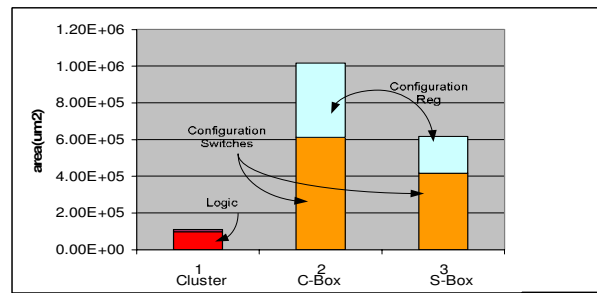


Fig-9 Area Distribution of Add-Subtract cluster

Conclusion

The elements are arranged in a proposed array with a mesh of reconfigurable interconnects. The re-configurability of the array permits mapping a number of DWT and filtering calculations used in video coding. In this paper, two DWT calculations were implemented on the RA. It was demonstrated that the array provides approximately a 31 to 38% reduction in power consumption over FPGA and an improvement of about 20 to 40% in timing. These figures show that the proposed array architectures provide a good compromise between hardwired ASICs and generic FPGAs in terms of flexibility, area, timing and power consumption when targeting a domain-specific application.

References

- [1] Bryant I., Tanurhan Y., "The Actel Embeddable FPGA Core", Actel Corporation, 2001
- [2] eASIC, "eASIC 0.13um Core", www.easic.com
- [3] Integer fast wavelet transform and its VLSI implementation or low power applications Dang, P.P.; Chau, P.M.; Signal Processing Systems, 2002. (SIPS '02).
- [4] Low power embedded extension algorithm for lifting-based Discrete wavelet transform in JPEG2000 Tan, K.C.B.; Arslan, T.; Electronics Letters, Volume: 37
- [5] K. Andra. C. Charabarti and T. Acharya, "A VLSI architecture for the lifting based forward and inverse wavelet transform", IEEE transaction on signal processing
- [6] D Le Gall and A. Tabatabai, "Sub-ban coding of digital images using symmetric short kernel filters and arithmetic coding", IEEE Intl conference on Acoustics, 1988
- [7] Rose J., Brown S., "Flexibility of interconnection structures for field-programmable gate arrays", Solid-State Circuits, IEEE Journal
- [8] Xilinx, The Programmable Logic Data Book, Xilinx Inc.
- [9] M. Anthoni, M Barlaud, P Mathieu, I Daubechies, "Image coding using wavelet transform", IEEE transaction on image processing, Col. 1, 1992
- [10] W. Sweldens, "The lifting scheme: A new philosophy in biorthogonal wavelet", Proceedings of SPIE,
- [11] ISO/IEC15444-1, "An information Technology-JPEG-2000 image coding system-Part 1: Core design System" <http://www.jpeg.org/JPEG2000.html,2000>.