Design of Modified Distributive Arithmetic Based DWT Processor for image and signal analysis

Ms.Safarfan P K
Sree Narayana Gurukulam College of Engineering,Kadayiruppu,Kolenchery

Ms.Seena George
Sree Narayana Gurukulam College of Engineering,Kadayiruppu,Kolenchery

Abstract

The Discrete Wavelet Transform (DWT) is being increasingly used for image coding. This is because the DWT can decompose the signals into different sub-bands with both time and frequency information. Complexity of DWT is always high due to large number of arithmetic operations. In this work a modified Distributive Arithmetic based DWT architecture is proposed for various applications like image compression, ECG signal analysis etc. This design utilizes fewer resources and thus consumes less power than the reference design.

Keywords— Discrete Wavelet Transforms (DWT), Distributive Arithmetic (DA), Poly-phase structure, and convolution

1. Introduction

Study has shown that the 90% of total volume of data in internet access consists of image and video related data. Image and video in their raw (uncompressed) form requires huge storage space. Such raw data needs large transmission bandwidth for the transmission over the network. Hence, lots of researches have been conducted in the field of data compression system. However, in this modern internet age, the demand for data transmission and the data storage are increasing. In this concern, data compression and reconstruction is the only option to relieve the network congestion. The compression technique reduces the size of data, which in turn requires less bandwidth and less transmission time and related cost. There are algorithms developed for the data compression such as: Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Walsh Hadamard Transform (WHT), etc.In this work, a reliable, high speed, low power DWT processor is designed which can be used as a co-processor for image compression and ECG signal analysis.

The Discrete Wavelet Transform (DWT) is being increasingly used for image coding. This is because the DWT can decompose the signals into different sub-bands with both time and frequency information. It also supports features like progressive image transmission, compressed image manipulation, and region of interest coding. Recently several VLSI architectures have been proposed to realize single chip designs for DWT. Traditionally, such algorithms are implemented using programmable DSP chips for low-rate applications, or VLSI application specific integrated circuits (ASICs) for higher rates. In wavelet transforms, the original signal is divided into frequency resolution and time resolution contents. The decomposition of the image using 2-level DWT is shown in Figure 1.

DWT is a multi-resolution transform and variable compression can be easily achieved. The main disadvantage of DWT is requirement of large computational resources. In case of stand-alone DWT, to achieve higher compression, higher level of DWT decomposition stages need to be considered. This will increase the computational complexity, and, degrades the reconstruction quality.

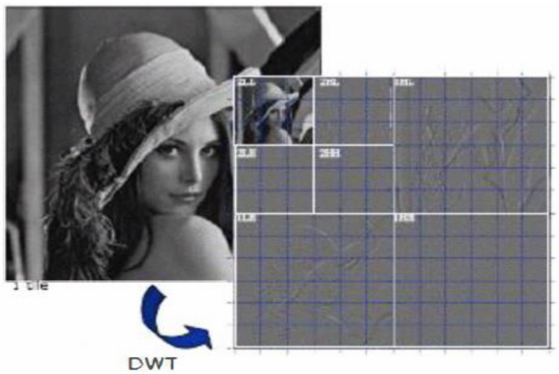


Figure 1. Decomposition of image

In this paper we describe a modified distributive arithmetic based the DWT architecture for image compression. We make maximal utilization of the lookup table (LUT) architecture by reformulating the wavelet transform computation in accordance with the distributed arithmetic algorithm. Moreover, distributed arithmetic is suitable for low power portable applications because it allows replacement of costly multipliers with shifts and look-up tables. Finally, this paper describes implementations of modified DA based DWT architectures for ECG signal enhancement.The main objective of this research work is to compare modified DA based DWT architecture with conventional DA based DWT architecture for image compression and extend modified DA based architecture to other applications like ECG signal analysis. This paper is organised as follows. Section 2 describes in brief the concept of traditional DWT and convolution-based filter architectures. In section 3, a detailed analysis of the modified DA algorithm and filter architecture will be addressed. This is followed by results and the conclusions. Results are presented in section 4 and section 5 is the conclusion and future enhancement.

2. DWT architecture

Image consists of pixels that are arranged in two dimensional matrix, each pixel represents the digital equivalent of image intensity. In spatial domain adjacent pixel values are highly correlated and hence redundant. In order to compress images, these redundancies existing among pixels needs to be eliminated. DWT processor transforms the spatial domain pixels into frequency domain information that are represented in multiple sub-bands, representing different time scale and frequency points. Human visual system is very much sensitive to low frequency and hence, the decomposed data available in the lower sub-band region and is selected and transmitted, information in the higher sub-bands regions are rejected depending upon required information content. In order to extract the low frequency and high frequency sub bands DWT architecture shown in figure below is used. As shown in the figure, input image consisting rows and columns are transformed using high pass and low pass filters. The filter coefficients are predefined and depend upon the wavelets selected. First stage computes the DWT output along the rows, the second stage computes the DWT along the column achieving first level decomposition. Low frequency sub-bands from the first level decomposition is passed through the second level and third level of filters to obtain multiple level decomposition as shown in Figure 2.

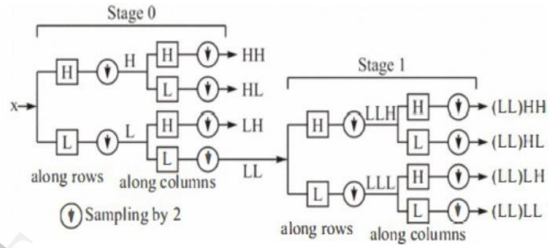


Figure 2. DWT Architecture

The process of decomposing a signal *x* into approximation and detail parts can be realized as a filter bank followed by down-sampling (by a factor of *2*) as shown in Figure 2. The impulse responses *h[n]* (low-pass filter) and *g[n]* (high-pass filter) are derived from the scaling function and the mother wavelet. This gives a new interpretation of the wavelet decomposition as splitting the signal *x* into frequency bands. In hierarchical decomposition, the output from the low-pass filter *h* constitutes the input to a new pair of filters. This results in a multilevel decomposition. The maximum number of such decomposition levels depends on the signal length. For a signal of size *N*, the maximum decomposition level is $\log_2(N)$.

There are several architectures for realizing the DWT shown in Figure 2. Most popular one is the DA-DWT scheme, as it consumes fewer resources and has high through put. DA-DWT architecture based on pipelining and parallel processing logic is realized and implemented on FPGA [12]. In this work, a modified DA-DWT architecture is designed based on the work reported in [12]. A control logic designed loads the input data into the FPGA from the external memory, LUT contents are read out based on the input samples as address to LUT. After 8 clock cycles of initial latency, DWT outputs are computed every clock cycle. A detailed discussion of the proposed architecture is presented in section 3.

The memory based approach provides an efficient way to replace multipliers by small ROM tables such that the DWT filter can attain high computing speeds with a small silicon area as shown in Figure 3. Traditionally, multiplication is performed using logic elements such as adders, registers etc. However, multiplication of two n-bit input variables can be performed by a ROM table of size of 2^{2n} entries. Each entry stores the pre-computed result of a multiplication. The speed of the ROM lookup table is faster than that of hardware multiplication if the look-up table is stored in the on-chip memory. In DWT, one of the input variables in the multiplier can be fixed. Therefore, a multiplier can be realized by 2^n entries of ROM. Distributed arithmetic implementation of the Daubechies 8-tap wavelet FIR filter consists of an LUT, a cascade of shift registers and a scaling accumulator [12].

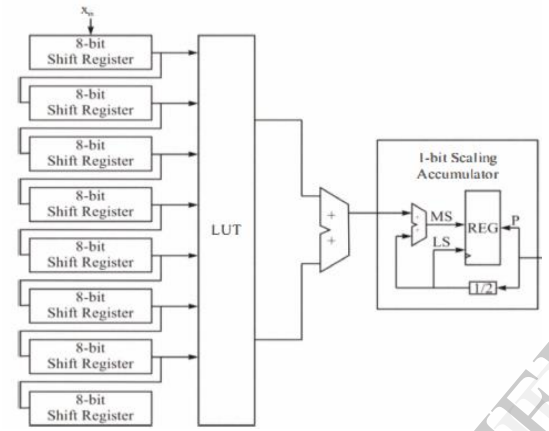


Figure 3.Distributed arithmetic

To speed up the process we can go for the parallel implementation of the Distributive Arithmetic (DA). The structure is as shown in the Figure 4. In parallel implementation, we divide the input data into even samples and the odd samples based on their position. Even we can split the filter coefficients into even and odd samples. So, the even samples convolve with the even and odd filter coefficients and at the same time the odd samples also convolve with the same coefficients. So, by the same time we are getting the result for both even and odd samples of input. This scheme reduces the memory size to half. This increases the through put as the input samples are simultaneously used to read the data from two LUTs and hence speed is increased.

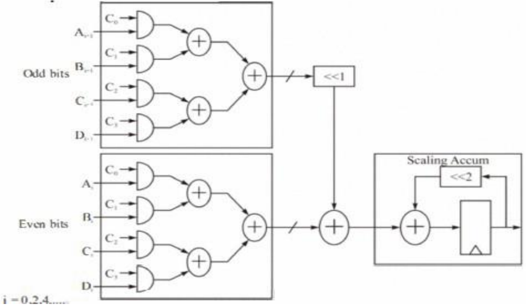


Figure 4. Parallel implementation of DA

In order to further increase the speed and reduce the area, the LUT can be further split into four stages, and can be accessed by the input values for data read.

3. Modified DA-DWT architecture

The modified DA-DWT architecture shown in Figure 5 consists of four LUTs, each of the LUTs are accessed by the even and odd samples of input matrix simultaneously. Odd and even input samples are divided into 4 bits of LSB and 4 bits of MSB, each 4-bit data read the content of four different LUTs that consist of partial products of filter values computed and stored as per the DA logic. Input samples are split into even and odd in the first stage, the data is further loaded sequentially into the serial in serial out shift registers, top four shift register store MSB bits and bottom four shift register stores the LSB bits. It requires 40 clocks cycles to load the shift register contents. At the end of 40th clock cycle, the control logic configures the shift register as serial in parallel out, thus forming the address for the LUT. The partial products stored in the LUT are read simultaneously from all the four LUTS and are accumulated with previous values available across the shift register in the output stage. The output stage consisting of adders, accumulators and right shift registers are used to accumulate the LUT contents and thus compute the DWT output.

4. Results and Discussion

HDL model for the proposed architecture is developed using Verilog. The developed model is simulated using test bench. Simulation results of Modified Distributive Arithmetic Based DWT processor for Image Compression are shown in figure 5.The HDL model is synthesized using Xilinx ISE targeting Spartan III-pro FPGA. The proposed design is synthesized and the report is generated. The synthesis results obtained for modified DA-DWT processor for image compression is presented in Table 1.The proposed design occupies only 30% of the total slices on FPGA, thus the proposed architecture reduces the area by 50%. Simulation results of Modified Distributive Arithmetic Based DWT processor for ECG signal enhancement is shown in figure 6. The proposed design consumes less power compared to the earlier designs when synthesized for various applications. A comparison between power consumed by conventional and proposed architecture is done in Table 3.

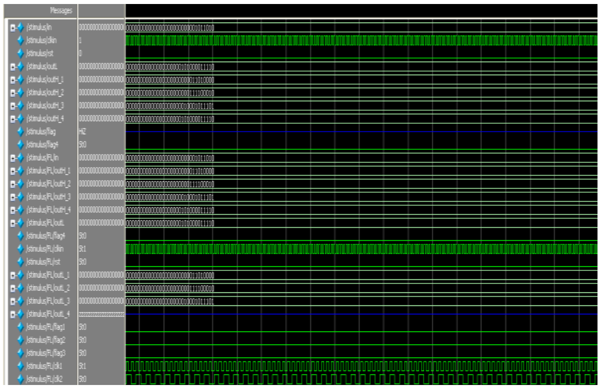


Figure 5. Simulation result- Image compression using modified DA-DWT architecture

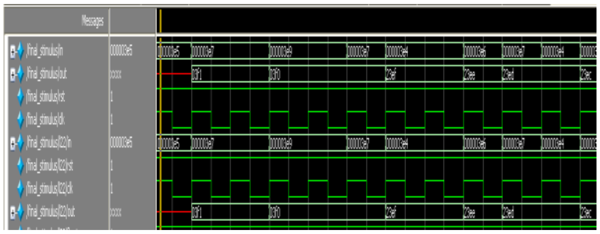
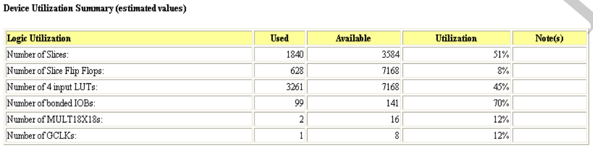


Figure 6.Simulation result ECG Signal Analysis using modified DA-DWT architecture

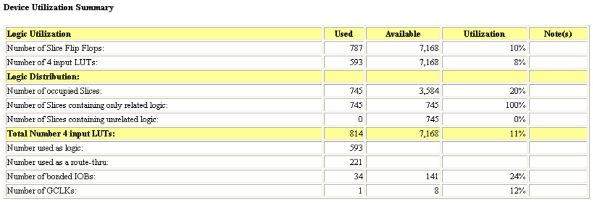
Table 1.Synthesis Results for image compression using modified DA-DWT architecture



Device Utilization Summary (estimated values)

Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slices:	1840	3584	51%	
Number of Slice Flip Flops:	628	7168	8%	
Number of 4 input LUTs:	3261	7168	45%	
Number of bonded IOBs:	99	141	70%	
Number of MULT18X18s:	2	16	12%	
Number of GCLKs:	1	8	12%	

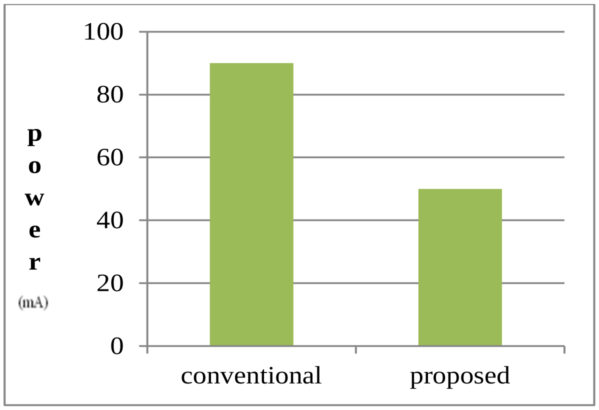
Table 2.Synthesis Results for ECG signal analysis using modified DA-DWT architecture



Device Utilization Summary

Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops:	787	7,168	10%	
Number of 4 input LUTs:	593	7,168	8%	
Logic Distribution:				
Number of occupied Slices:	745	3,584	20%	
Number of Slices containing only related logic:	745	745	100%	
Number of Slices containing unrelated logic:	0	745	0%	
Total Number 4 input LUTs:	814	7,168	11%	
Number used as logic:	593			
Number used as a route-thru:	221			
Number of bonded IOBs:	34	141	24%	
Number of GCLKs:	1	8	12%	

Table 3.Power Consumption



5. Conclusion

The Discrete Wavelet Transform provides a multi resolution representation of images. The transform has been implemented using filter banks. For the design, based on the constraints the area, power and timing performance were obtained. Based on the application and the constraints imposed, the appropriate architecture can be chosen architecture, with modified DA technique was implemented. It is seen that, in applications, which require low area, power consumption, and high throughput, e.g., real-time applications, the poly-phase with DA architecture is more suitable. First, the code was written in Verilog HDL and is synthesized using Xilinx ISE targeting Spartan III-pro FPGA. This architecture enables fast computation of DWT with parallel processing. It has low memory requirements and consumes low power. Even though wavelet transform had been used profusely for image compression tasks, the choice is not the ideal one. The partial reconstruction error from wavelet coefficients is an order of magnitude higher than the ideal error rate for many critical applications. Image compression can be carried in the curvelet domain—a better choice compared to wavelets, at least theoretically, since the reconstruction error rate with curvelet coefficients is of the same asymptotic order as that of the ideal error rate

6. References

[1] M.Nagabushanam,Cyril Prasanna Raj P,S.Ramachandran,"Design & FPGA Implementation of Modified DA based DWT-IDWT Processor for Image Compression", IEEE transactions on very large scale integration (VLSI) systems,vol.3,Aug 2011.

[2]Nagabhushanam,Cyril Prasanna Raj P,Ramachandran,"Design &implementation of Parallel & pipelined DA based DWT IP core"

[3] David S. Taubman, Michael W. Marcellin - JPEG 2000 – Image compression, fundamentals, standards and practice", Kluwer academic publishers, Second printing - 2002.

[4] G. Knowles, "VLSI Architecture for the Discrete Wavelet Transform,"Electronics Letters, vo1.26, pp. 1184-1185,1990.

[5] M, Vishwanath, R. M. Owens, and M. 1. Irwin, "VLSI Architectures for the Discrete Wavelet Transform," IEEE Trans. Circuits And Systems II, vol. 42, no. 5, pp. 305-316, May. 1995.

[6] AS. Lewis and G. Knowles, "VLSI Architectures for Electron Letter, vo1.27, pp. 171-173, Jan 1991.

[7] K.K. Parhi and T. Nishitani "VLSI Architecture for Discrete Wavelet Transform", IEEE Trans. VLSI Systems, vol. 1, pp. 191-202, June 1993

[8] David S. Taubman, Michael W. Marcellin – JPEG 2000 – Image compression, fundamentals,standards and practice", Kluwer academic publishers, Second printing – 2002.

[9] G. Knowles, "VLSI Architecture for the Discrete Wavelet Transform," Electronics Letters, vol.26, pp. 1184-1185, 1990.

[10] M, Vishwanath, R. M. Owens, and M. J. Irwin, "VLSI Architectures for the Discrete Wavelet Transform," IEEE Trans. Circuits And Systems II, vol. 42, no. 5, pp. 305-316, May. 1995.

[11] Majid Rannani and Rajan Joshi, "An Overview of the JPEG2000 Still Image Compression Standard", Signal Processing, Image Communication, vol. 17, pp. 3-48, 2002.

[12] K. Seth and S. Srinivasan, "VLSI Implementation of 2D DWT/IDWT Cores using 9/7-tap filter banks based on Non-Expansive Symmetric Extension Scheme" proceedings of ASPDAC/VLSI Design 2002, Bangalore, India, 7-11 January 2002.