ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 2, Issue 12, December- 2013

A NOVEL ARCHITECTURE FOR AN EFFICIENT IMPLEMENTATION OF 2D-DWT

A.Sairamya[1] , Kakani Bhargav[2], B.Alekhya[3]

[1]Student (M.Tech), VLSI (SD), Department of ECE, Avanthi Institute of Engineering &Technology ,
[2]Asst.Professor, Department of ECE, Avanthi Institute of Engineering &Technology,
[3]Student (M.Tech), VLSI (SD), Department of ECE, Avanthi Institute of Engineering &Technology ,

Abstract

The wavelet transform (WT) has gained widespread acceptance in signal processing and image compression. Because of their inherent multi-resolution nature, wavelet-coding schemes are especially suitable for applications where scalability and tolerable degradation are important. To use the wavelet transform for image processing we must implement a 2D version of the analysis. In this paper, pipeline architecture for fast computation of the 2D DWT low latency is proposed. The low latency is achieved by proper designing of two 1-D DWT filtering processes and also efficiently transferring the data between the two 1-D DWT filters.

Index Terms: *Discrete wavelets transform (DWT), lifting scheme, pipeline, VLSI architecture.*

1. INTRODUCTION

Very-large-scale integration (VLSI) is the process of creating integrated circuits by combining thousands of transistor-based circuits into a single chip. VLSI began in the 1970s when complex semiconductor and communication technologies were being developed. The microprocessor is a VLSI device [2]. The term is no longer as common as it once was, as chips have increased in complexity into the hundreds of millions of transistors [9].

The first semiconductor chips held one transistor each. Subsequent advances added more and more transistors, and, as a consequence, more individual functions or systems were integrated over time. The first integrated circuits held only a few devices, perhaps as many as ten diodes, transistors, resistors and capacitors, making it possible to fabricate one or more logic gates on a single device. Now known retrospectively as "small-scale integration" (SSI), improvements in technique led to devices with hundreds of logic gates, known as large-scale integration (LSI), i.e. systems with at least a thousand logic gates. Current technology has moved far past this mark and today's microprocessors have many millions of gates and hundreds of millions of individual transistors.

At one time, there was an effort to name and calibrate various levels of large-scale integration above VLSI. Terms like Ultra-large-scale Integration (ULSI) were used. But the huge number of gates and transistors available on common devices has rendered such fine distinctions moot [4]. Terms suggesting greater than VLSI levels of integration are no longer in widespread use. Even VLSI is now somewhat quaint, given the common assumption that all microprocessors are VLSI or better.

2. VERILOG HDL

Verilog HDL is a hardware description language that can be used to model a digital system at many levels of abstraction ranging from the algorithmic-level to the gate-level to the switch-level [5]. The complexity of the digital system being modeled could vary from that of a simple gate to a complete electronic digital system, or anything in between. The digital system can be described hierarchically and timing can be explicitly modeled within the same description.

The Verilog HDL language includes capabilities to describe the behavior-al nature of a design, the dataflow nature of a design, a design's structural composition, delays and a waveform generation mechanism including aspects of response monitoring and verification, all modeled using one single language [6]. In addition, the language provides a programming language interface through which the internals of a design can be accessed during simulation including the control of a simulation run.

The language not only defines the syntax but also defines very clear simulation semantics for each language construct. Therefore, models written in this language can be verified using a Verilog simulator. The language inherits many of its operator symbols and constructs from the C programming language. Verilog HDL provides an extensive range of modeling capabilities, some of which are quite

difficult to comprehend initially [1]. However, a core subset of the language is quite easy to leam and use. This is sufficient to model most applications.

The major capabilities of Verilog hardware description is given below:

- Primitive logic gates, such as and, or and nand gates, are built-in into the language.
- Flexibility of creating a user-defined primitive (UDP). Such a primitive could either be a combinational logic primitive or a sequential logic primitive.
- Switch-level modeling primitive gates, such as pmos and nmos, are also built-in into the language.
- Explicit language constructs are provided for specifying pin-to-pin delays, path delays and timing checks of a design.
- A design can be modeled in three different styles or in a mixed style. These styles are: behavioral style - modeled using procedural constructs; dataflow style - modeled using continuous assignments; and structural style - modeled using gate and module instantiations.
- There are two data types in Verilog HDL; the net data type and the register data type. The net type represents a physical connection between structural elements while a register type represents an abstract data storage element.
- Figure.1 shows the mixed-level modeling capability of Verilog HDL, that is, in one design; each module may be modeled at a different level.

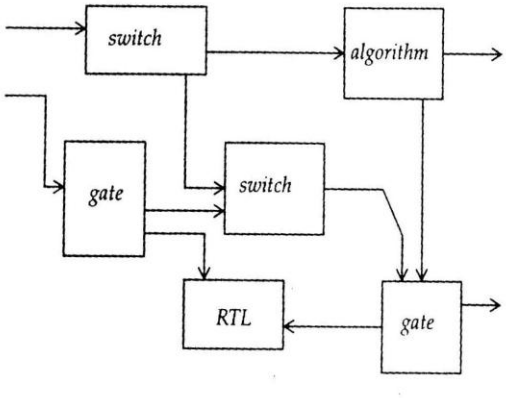

Fig-1 Mixed-level Modeling

2.1 SYNTHESIS DESGIN

Synthesis is the process of constructing a gate level netlist from a register-transfer level model of a circuit described in Verilog HDL. Figure.2-2 shows such a process. A synthesis system may as an intermediate step, generate a netlist that is comprised of register-transfer level blocks such as flip-flops, arithmetic-logic-units, and multiplexers, interconnected by wires. In such a case, a second program called the RTL module builder is necessary. The purpose of this builder is to build, or acquire from a library of predefined components, each of the required RTL blocks in the user-specified target technology.

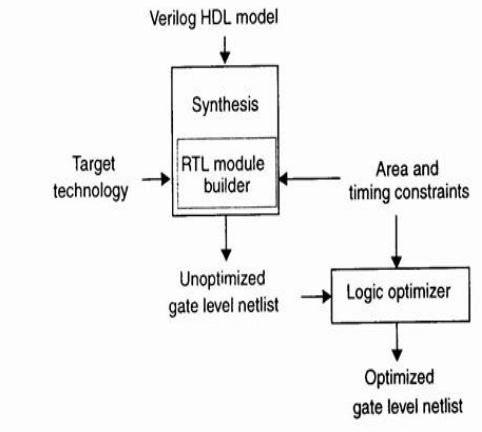

Fig-2 Synthesis Process

The above figure shows the basic elements of Verilog HDL and the elements used in hardware. A mapping mechanism or a construction mechanism has to be provided that translates the Verilog HDL elements into their corresponding hardware elements as shown in figure-3

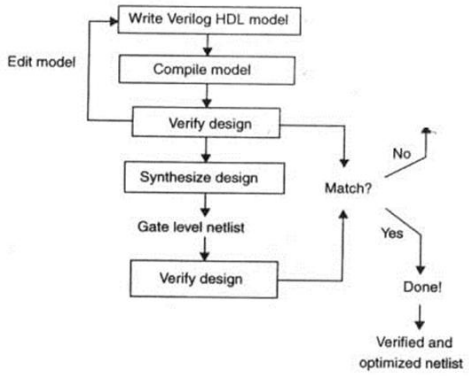

Fig-3 Typical design process

3. DISCRETE WAVELET TRANSFORM

A 'wavelet' is a small wave which has its energy concentrated in time. It has an oscillating wavelike characteristic but also has the ability to allow simultaneous time and frequency analysis and it is a suitable tool for transient, non-stationary or time-varying phenomena.

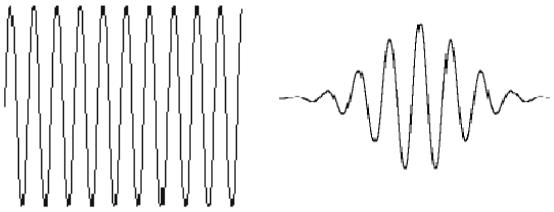

Fig-4 (a) Wave (b) Wavelet

Wavelet Transform uses a set of damped oscillating functions known as wavelet basis. WT in its continuous (analog) form is represented as CWT. CWT with various deterministic or non-deterministic bases is a more effective representation of signals for analysis as well as characterization. Continuous wavelet transform is powerful in singularity detection. A discrete and fast implementation of CWT (generally with real valued basis) is known as the standard DWT (Discrete Wavelet Transform).With standard DWT, signal has a same data size in transform domain and therefore it is a non-redundant transform. A very important property was Multi-resolution Analysis (MRA) allows DWT to view and process.

The wavelet analysis procedure is to adopt a wavelet prototype function, called an 'analyzing wavelet' or 'mother wavelet'. Temporal analysis is performed with a contracted, high frequency version of the prototype wavelet, while frequency analysis is performed with a dilated, low frequency version of the same wavelet. Mathematical formulation of signal expansion using wavelets gives Wavelet Transform (WT) pair, which is analogous to the Fourier Transform (FT) pair. Discrete-time and discrete-parameter version of WT is termed as Discrete Wavelet Transform (DWT).

In this we are designing the pipelining architecture of 2-D Discrete wavelet Transform (DWT) with a low latency proposed.

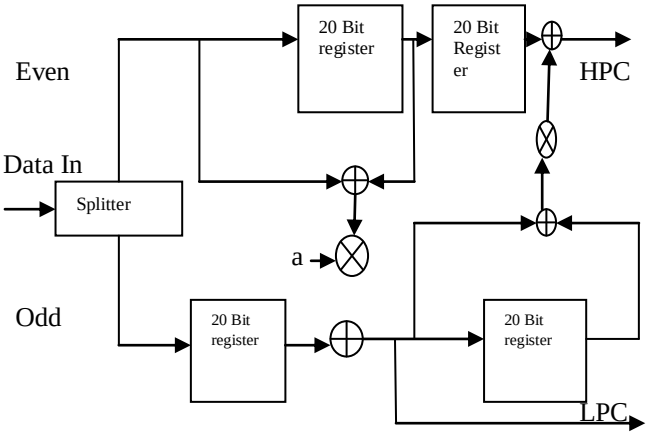

Fig-5 Proposed Architecture

In the initial stage the input data given is split into even and odd positions. Its given to 20-bit register where it is summation done with the predict data which is given to the other side of data analysis. The update data is done summation with output data obtained from the 20-bit register. At the same time parallel data is received as coefficients from the either side in terms of Low-pass coefficients and High-pass coefficients.

The proposed architecture explains the data transmission efficiently in 1-D DWT same time. The data transmission is been done in XILINX tool.

4. SYNTHESIS RESULTS

This plot explains about the schematic output of the design architecture

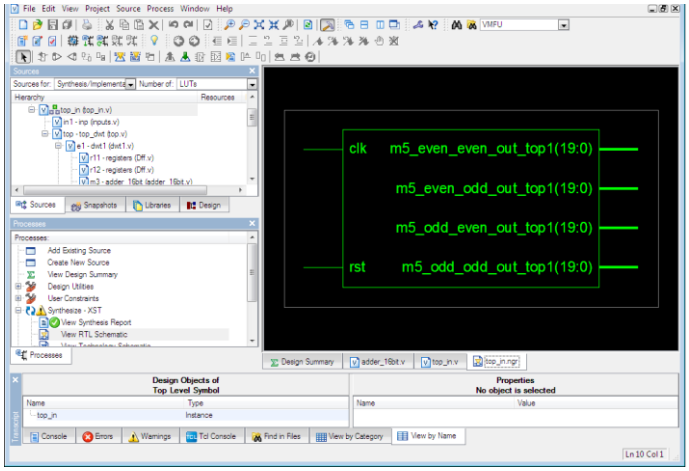

Fig-6 Schematic of Architecture

This plot explains the internal design of architecture in terms of RAMs, counters, adders, comparators and flip flops used.

Logic Utilization	Used
Number of RAMs	6
Number of 4x20-bit dual-port distributed RAM	4
Number of8x20-bit dual-port distributed RAM	1
Number of 16x20-bit ROM	1
Number of Adders/Sub tractors	15
Number of 20-bit adder	15
Number of Counters	7
Number of 3-bit up counter	4
Number of 32-bit up counter	1
Number of 4-bit up counter	2
Number of Registers	168
Number of Flip-Flops	168
Number of Comparators	9
Number of 3-bit comparator greater	2
Number of 3-bit comparator less equal	4
Number of 4-bit comparator greater	1
Number of 4-bit comparator less equal	2

Fig-7 Advanced HDL Design Report

This plot explains about the 1-D plot proposed architecture in graphical representation

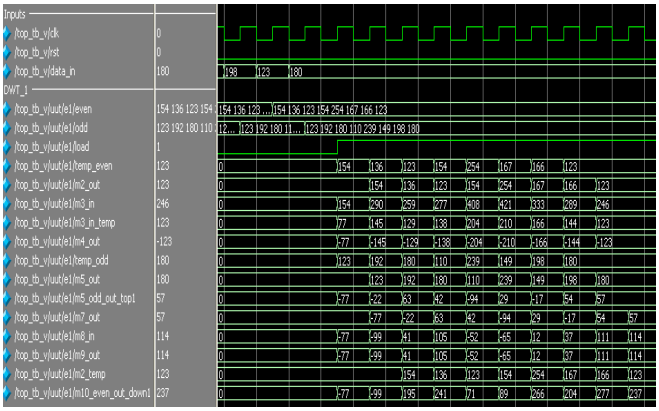

Fig-8 Simulation result of 1-D DWT both in High and Low pass coefficients.

This plot explains about the 2-D plot of proposed architecture in graphical representation.

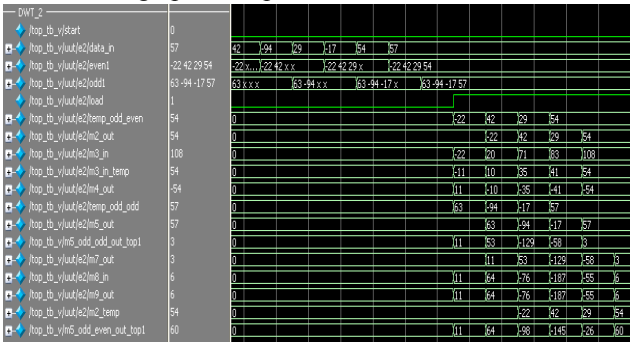

Fig-8 Simulation result of 2-D DWT both in High and Low pass coefficients.

5. CONCLUSION

In this paper, pipeline architecture for fast computation of the 2-D DWT with a low latency is proposed. The low latency is achieved by proper designing of two 1-D DWT filtering processes and also efficiently transferring the data between the two 1-D DWT architectures. This architecture is simulated, synthesized and implemented by VERILOG language using XILINX ISE Tool. We have proposed a novel architecture for the 1- and 2-D DWTs. The modified one lifting step circuit can work within three pipelining stages with fewer registers, and the critical path delay. For the 2-D DWT architecture, only the temporal buffer with $4N$ size is used in the column filter. A detailed analysis is performed to compare the proposed architectures with other previous architectures in terms of hardware complexity, critical path delay, storage size, computation time, and throughput. According to the results, the proposed architecture can achieve high speed with lower hardware complexity and smaller storage size. We can further extend

the implementation for proposed architecture using different techniques.

REFRENCES

[1] [1] G. Xing, J. Li, and Y. Q. Zhang, "Arbitrarily shaped video-object coding by wavelet," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 10, pp. 1135–1139, Oct. 2001.

[2] S. C. B. Lo, H. Li, and M. T. Freedman, "Optimization of wavelet decom-position for image compression and feature preservation," *IEEE Trans. Med. Imag.*, vol. 22, no. 9, pp. 1141–1151, Sep. 2003.

[3] K. K. Parhi and T. Nishitani, "VLSI architecture for discrete wavelet transforms," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 1, no. 2, pp. 191–202, Jun. 1993.

[4] P. Wu and L. Chen, "An efficient architecture for two-dimensional discrete wavelet transform," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 4, pp. 536–545, Apr. 2001.

[5] W. Sweldens, "The new philosophy in biorthogonal wavelet construc-tions," in *Proc. SPIE.*, 1995, vol. 2569, pp. 68–79.

[6] I. Daubechies and W. Sweldens, "Factoring wavelet transform into lifting steps," *J. Fourier Anal. Appl.*, vol. 4, no. 3, pp. 245–267, Mar. 1998.

[7] J. M. Jou, Y. H. Shiau, and C. C. Liu, "Efficient VLSI architectures for the biorthogonal wavelet transform by filter bank and lifting scheme," in *Proc. IEEE ISCAS*, May 2001, vol. 2, pp. 529–532.

[8] G. Shi, W. Liu, and L. Zhang, "An efficient folded architecture for lifting-based discrete wavelet transform," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 56, no. 4, pp. 290–294, Apr. 2009.

[9] B. F. Wu and C. F. Lin, "A high-performance and memory-efficient pipeline architecture for the 5/3 and 9/7 discrete wavelet transform of JPEG2000 codec," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 12, pp. 1615–1628, Dec. 2005.

[10] Y. K. Lai, L. F. Chen, and Y. C. Shih, "A high-performance and memory-efficient VLSI architecture with parallel scanning method for 2-D lifting-based discrete wavelet transform," *IEEE Trans.*

[11] pp. P.-C. Tseng, C.-T. Huang, and L.-G. Chen, "Generic RAM-based architecture for two dimensional discrete wavelet transform with line-based method," in *Proc. Asia-Pacific Conf. Circuits Syst.*, 2002, vol. 2, 363–366.

[12] C. Xiong, J. Tian, and J. Liu, "Efficient architectures for two-dimensional discrete wavelet transform using lifting scheme," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 607–614, Mar. 2007.

[13] H. Liao, M. K. Mandal, and B. F. Cockburn, "Efficient architectures for 1-D and 2-D lifting-based wavelet transforms," *IEEE Trans. Signal Process.*, vol. 52, no. 5, pp. 1315–1326, May 2004.

[14] C.-Y. Xiong, J.-W. Tian, and J. Liu, "A note on 'flipping structure: An efficient VLSI architecture for lifting-based discrete wavelet trans-form'," *IEEE Trans. Signal Process.*, vol. 54, no. 5, pp. 1910–1916, May 2006.

[15] C. Cheng and K. K. Parhi, "High-speed VLSI implement of 2-D discrete wavelet transform," *IEEE Trans. Signal Process.*, vol. 56, no. 1, pp. 393– 403, Jan. 2008.

[16] B. K. Mohanty and P. K. Meher, "Throughput-scalable hybrid-pipeline architecture for multilevel lifting 2-D DWT of JPEG 2000 coder," in
Proc. IEEE Int. Conf. Appl.-Specific Syst., Archit. Processors, 2008, pp. 305–309.

[17] J. Song and I.-C. Park, "Novel pipelined DWT architecture for dual-line scan," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2009, pp. 373–376.

[18] Z. G. Wu and W. Wang, "Pipelined architecture for FPGA implementation of lifting-based DWT," in *Proc. Int. Conf. Elect. Inform. Control Eng.*, 2011, pp. 1535–1538.

[19] B. K. Mohanty and P. K. Meher, "Efficient multiplierless designs for 1-D DWT using 97 filters based on distributed arithmetic," in *Proc. Int. Symp. Integr. Circuits*, 2009, pp. 364–367.

BIOGRAPHIES

A.Sairamya pursuing M.Tech in specialization of VLSI at Avanthi Institute of Engineering Technology in Department of Electronics and Communication Engineering.

K.Bhargav completed M.Tech and working as an Assistant Professor in Avanthi Institute of Engineering &Technology in Department of Electronics and communication Engineering.