Impact Factor (SJIF): 5.301

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444

Volume 5, Issue 3, March-2018

VLSI ARCHITECTURE FOR HIGH THROUGHPUT COMPUTATION OF MULTILEVEL HAAR WAVELET TRANSFORM

S.Sahana[1], M.Vijayalakshmi[2], K.V.Sravani[3]

[1,2,3]*Electronics and Communication Engineering, Panimalar Institute of Technology*

Abstract — **This paper presents a high precision low area lifting based architecture for the unified implementation of both lossy and lossless 3D multi-level Discrete Wavelet Transform (DWT) using Haar Transformer. The proposed system is parallel-pipelined, and resource is shared between the lossy and lossless modes, producing a throughput of 2 outputs/clock and achieving a high speed and low area solution. The data width of the design is taken as 20 bits to reach a high PSNR value for multi-level 3D DWT. Targeting a portable and real-time solution, the proposed architecture was successfully implemented on Xilinx Virtex-5 series Field Programmable Gate Array (FPGA), achieving a clock speed of 290 MHz with a power consumption of 467 mW at 200 MHz clock frequency. The design has also been implemented in UMC 90 nm CMOS technology, which consumes 329 mW power at 200 MHz clock frequency. The proposed solution may be configured as lossless compression, in the field of 3D image compression system, according to the necessity of the user.**

Keywords-Compression, Lifting, Latency, Haar Transformer, Xilinx, FPGA.

I. . INTRODUCTION

The DWT can be divided into two categories - lossy and lossless DWT. The lossy DWT is mainly used in situations that demand a high compression ratio; thus, it is very appealing in military, HD satellite images [1], motion detection, network distribution and storage purposes. On the other hand, lossless transformation is used in medical imaging [2], digital negative (DNG) and some digital cameras for compressing the images. But as the coefficients of the lossy filter are real floating point numbers, the computational complexity of implementation is very high. Moreover, the lossy transform is irreversible, i.e. there is some loss in the image during this transformation. For lossless DWT, the amount of compression is considerably less compared to that of lossy DWT. Ideally, infinite PSNR can be achieved using lossless DWT, which is a necessity in medical imaging. In applications such as satellite imaging systems [3] [4], multispectral imaging [5] and telemedicine systems, lossy and lossless both types of compression are necessary, where less important data and image thumbnails can go through lossy compression and high-resolution images and medical data can be compressed using lossless DWT. Satellite Image enhancement is the technique which is most widely required in the field of image processing to improve visualization of the feature [1].The DWT is an important operation in many image processing applications such as image compression, bio-informatics, and texture discrimination. The DWT, standardized in forms like JPEG 2000 [6], has many desirable features such as the absence of blocking artefacts, higher resolution capability, in place computation, and better peak signal to noise ratio (PSNR) etc. Today, many medical imaging systems such as computed tomography (CT), positron emission tomography (PET), and magnetic resonance imaging (MRI), are generating a large amount of volumetric image data sets. Many modern applications need these datasets to process online or offline with different resolutions and features (region of interests, scaling, etc.). Part 1 and Part 2 of JPEG 2000 standard [7] encompass all these requirements mentioned above for 2-D image. For volumetric data sets, the JPEG committee decided to add part 10 in the JPEG 2000 standard, commonly known as JP3D [8]. The basic transform algorithm for JP3D [9].is 3-D DWT. It is an effective submodule in video coding, like Motion-JPEG, which is shown to be more precise than MPEG-4 standard [10]. The advent of 3-D and 4-D medical imaging system increased the necessity of 3-D volumetric image compression system [11]. Processing of 3D magnetic resonance (MR) images of brain through 3-D DWT to extract features for detection of Alzheimer's disease and mild cognitive impairment in subjects [12].

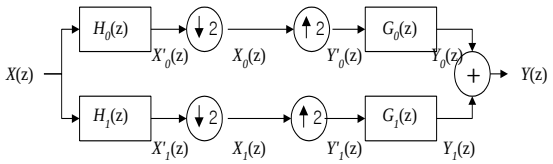
II. COMPRESSION

Compression is a process of reducing the number of data bits necessary for representing information, to properly utilize the available bandwidth and reduce Storage Space. There are two types of compression namely Lossless compression and Lossy compression.

2.1. Lossy compression: Data cannot be completely recovered after decompression, some information is lost for ever, gives more compression than lossless, discards "insignificant" data components.
2.2. Lossless compression: In lossless compression data can be completely recovered after decompression Recovered data is identical to original, exploits redundancy in data.

III. WAVELET TRANSFORM

There are two approaches to make a wavelet transform: Scaling function and wavelets (the dilation equation and wavelet equation) by mathematicians_Filter banks (low-pass filter and high-pass filter) by engineers. The two approaches produce same results, proved by Doublets. Wavelet Transform is a type of signal representation that can give the frequency content of the signal at a particular instant of time. Wavelet analysis has advantages over traditional Fourier methods in analysing physical situations where the signal contains discontinuities and sharp spikes.



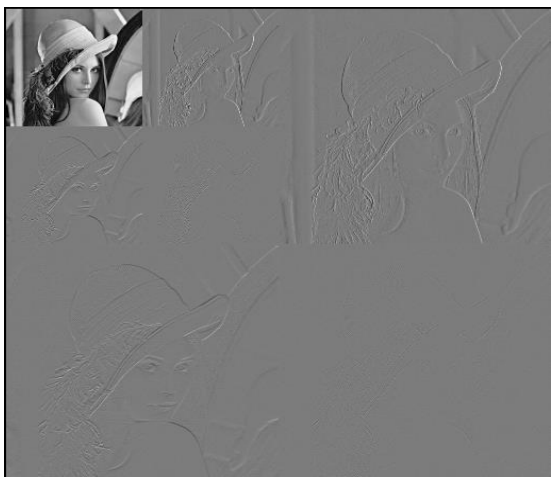
Wavelet Transform
(Ideal low-pass and high-pass filter bank)

3.1. DISCRETE WAVELET TRANSFORM:

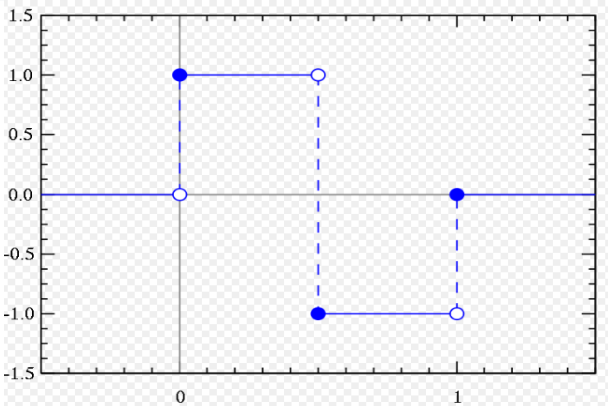
The discrete wavelet transform (DWT) has been developed as an efficient DSP tool for signal analysis, image compression, and even video compression [1]. There is much architecture proposed for the implementation of DWT. For the 1-D DWT, the architectures can be categorized into the convolution- based, lifting-based, and B-spline-based. The first one is to implement two-channel filter banks directly. The second one is to exploit the relationship of low pass and high pass filters for saving multipliers and adders. The third one can reduce the multipliers based on the B-spline factorization. The B-spline-based architectures could provide fewer multipliers while the lifting scheme fails to reduce the complexity.

3.1.1. HAAR WAVELET

In mathematics, the **Haar wavelet** is a sequence of rescaled "square-shaped" functions which together form a wavelet family or basis. Wavelet analysis is similar to Fourier analysis in that it allows a target function over an interval to be represented in terms of an orthonormal basis. The Haar sequence is now recognized as the first known wavelet basis and extensively used as a teaching example. The **Haar sequence** was proposed in 1909 by Alfred Haar. Haar used these functions to give an example of an orthonormal system for the space of square-integral functions on the unit interval [0, 1]. The study of wavelets, and even the term "wavelet", did not come until much later. As a special case of the Daubechies wavelet, the Haar wavelet is also known as **Db1**.The Haar wavelet is also the simplest possible wavelet. The technical disadvantage of the Haar wavelet is that it is not continuous, and therefore not differentiable. This property can, however, be an advantage for the analysis of signals with sudden transitions, such as monitoring of tool failure in machines.



Two iterations of the 2D Haar wavelet decomposition on the Lenna image. The original image is high-pass filtered, yielding the three detail coefficients sub images (top right: horizontal, bottom left: vertical, and bottom right: diagonal). It is then low-pass filtered and downscaled, yielding an approximation coefficients sub image (top left); the filtering process is repeated once again on this approximation image.



In mathematics, the **Haar wavelet** is a certain sequence of functions. It is now recognised as the first known wavelet. This sequence was proposed in 1909 by Alfred Haar. Haar used these functions to give an example of a countable orthonormal system for the space of square integral functions on the real line. The study of wavelets, and even the term "wavelet", did not come until much later. The Haar wavelet is also the simplest possible wavelet. The technical disadvantage of the Haar wavelet is that it is not continuous, and therefore not differentiable.

The Haar wavelet's mother wavelet function $\psi(t)$ can be described as

$$\psi(t) = \begin{cases} 1 & 0 \le t < 1/2, \\ -1 & 1/2 \le t < 1, \\ 0 & \text{otherwise.} \end{cases}$$

and its scaling function $\varphi(t)$ can be described as

$$\phi(t) = \begin{cases} 1 & 0 \le t < 1, \\ 0 & \text{otherwise.} \end{cases}$$

Wavelets are mathematical functions that were developed by scientists working in several different fields for the purpose of sorting data by frequency. Translated data can then be sorted at a resolution which matches its scale. Studying data at different levels allows for the development of a more complete picture. Both small features and large features are discernible because they are studied separately. Unlike the discrete cosine transform, the wavelet transform is not Fourier-based and therefore wavelets do a better job of handling discontinuities in data. The Haar wavelet operates on data by calculating the sums and differences of adjacent elements. The Haar wavelet operates first on adjacent horizontal elements and then on adjacent vertical elements.

The Haar transform is computed using:

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

There are different ways to include processors inside Xilinx FPGA for System-on-a-Chip (SOC): PowerPC hard processor core, or Xilinx Micro Blaze soft processor core, or user-defined soft processor core in VHDL/Verilog. In this work, The 32-bit Micro Blaze processor is chosen because of the flexibility. The user can tailor the processor with or without advance features, based on the budget of hardware. The advance features include memory management unit, floating processing unit, hardware multiplier, hardware divider, instruction and data cache links etc. The architecture overview of the system is shown in Figure 2. It can be seen that there are two different buses (i.e., processor local bus (PLB) and fast simplex link (FSL bus) used in the system [5-6]. PLB follows IBM core connect bus architecture, which supports high bandwidth master and slave

devices, provides up to 128- bit data bus, up to 64-bit address bus and centralized bus Arbitration. It is a type of shared bus. Besides the access overhead, PLB potentially has the risk of hardware/software incoherent due to bus arbitration. On the other hand, FSL supports point-to-point unidirectional communication. A pair of FSL buses (from processor to peripheral and from peripheral to processor) can form a dedicated high speed bus without arbitration mechanism. Xilinx provides C and assembly language support for easy access. Therefore, most of peripherals are connected to the processor through PLB; the DWT coprocessor is connected through FSL instead.

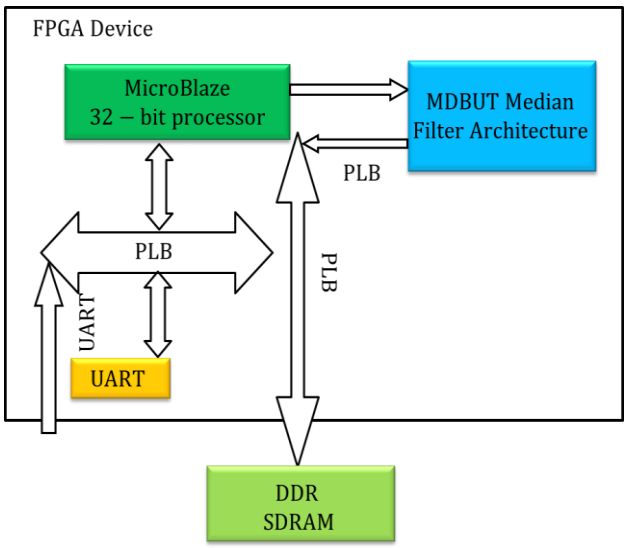


Fig-System Overview

The current system offers several methods for distributing the data. These methods are a UART, and VGA, and Ethernet controllers. The UART is used for providing an interface to a host computer, allowing user interaction with the system and facilitating data transfer. The VGA core produces a standalone real-time display. The Ethernet connection allows a convenient way to export the data for use and analysis on other systems. In our work, to validate the DWT coprocessor, an image data stream is formed using Visual Basic, then transmitted from the host computer to FPGA board through UART port.

In terms of its instruction-set architecture, Micro Blaze is very similar to the RISC-based DLX architecture described in a popular computer architecture book by Patterson and Hennessy. With few exceptions, the Micro Blaze can issue a new instruction every cycle, maintaining single-cycle throughput under most circumstances.

The Micro Blaze has a versatile interconnect system to support a variety of embedded applications. Micro Blaze's primary I/O bus, the Core Connect PLB bus, is a traditional system-memory mapped transaction bus with master/slave capability. A newer version of the Micro Blaze, supported in both Spartan-6 and Virtex-6 implementations, as well as the 7-Series, supports the AXI specification. The majority of vendor-supplied and third-party IP interface to PLB directly (or through an PLB to OPB bus bridge.) For access to local-memory (FPGA BRAM), Micro Blaze uses a dedicated LMB bus, which reduces loading on the other buses. User-defined coprocessors are supported through a dedicated FIFO-style connection called FSL (Fast Simplex Link). The coprocessor(s) interface can accelerate computationally intensive algorithms by offloading parts or the entirety of the computation to a user-designed hardware module.

The Micro Blaze processor is a 32-bit Harvard Reduced Instruction Set Computer (RISC)architecture optimized for implementation in Xilinx FPGAs with separate 32-bit instruction and data buses running at full speed to execute programs and access data from both on-chip and external memory at the same time. The backbone of the architecture is a single-issue, 3-stage pipeline with 32 general-purpose registers (does not have any address registers like the Motorola 68000 Processor), an Arithmetic Logic Unit (ALU), a shift unit, and two levels of interrupt. This basic design can then be configured with more advanced features to tailor to the exact needs of the target embedded application such as: barrel shifter, divider, multiplier, single precision floating-point unit (FPU), instruction and data caches, exception handling, debug logic, Fast Simplex Link (FSL) interfaces and others. This flexibility allows the user to balance the required performance of the target application against the logic area cost of the soft processor. The items in white are the backbone of the Micro Blaze architecture while the items shaded grey are optional features available depending on the exact needs of

the target embedded application. Because Micro Blaze is a soft-core microprocessor, any optional features not used will not be implemented and will not take up any of the FPGAs resources.
The Micro Blaze is a virtual microprocessor that is built by combining blocks of code called cores. Micro Blaze is an embedded soft core that includes the following features
• Thirty-two 32-bit general purpose registers.
• 32-bit instruction word with three operands and two addressing modes.
• Separate 32-bit instruction and data buses that conform to IBM's OPB (On-chip Peripheral Bus) specification.
• 32-bit address bus
• Single issue pipeline

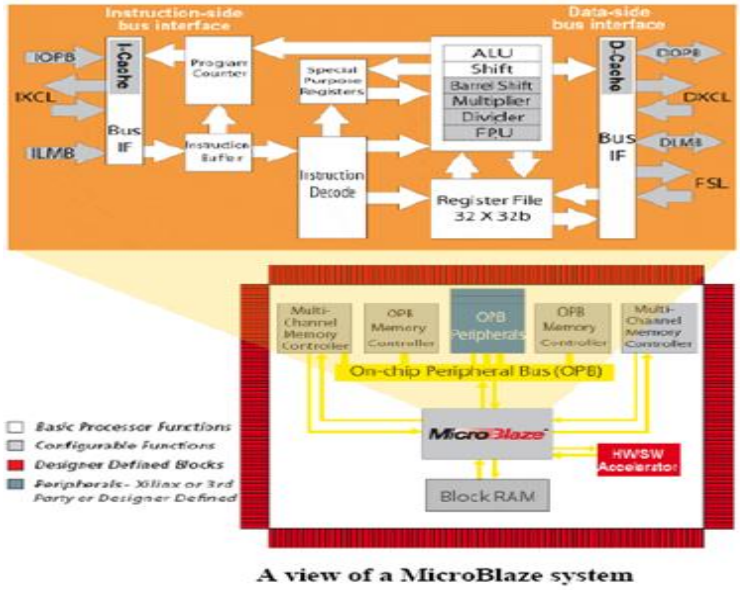


Fig: Micro Blaze system

IV. PROPOSED ARCHITECTURE

The Spartan-3E FPGA is embedded with the 90nm technology at the heart of its architecture. This reduces the die size and cost, increases manufacturing efficiency, and addresses a wider range of applications. You can integrate embedded processing, digital signal processing (DSP), and connectivity capabilities into Spartan-3E devices at no extra cost. These are supported with customized tools (ISE and EDK), JTAG probes, IP cores, design services, and training. The Spartan-3E diagram shown in allows users to easily migrate to different densities across multiple packages and supports 18 different single-ended and differential I/O standards.

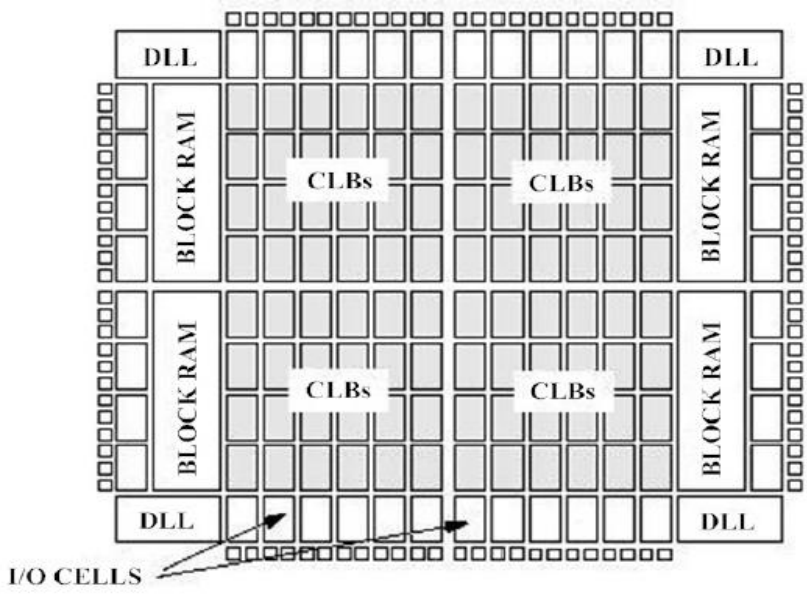


Fig- Spartan 3E Layout

The main advantages are High Speed Connectivity, High Performance DSP Solutions and Lowest Cost Embedded Processing Solutions.

4.1.1. High Speed Connectivity

System connectivity consists of physical parallel I/O interfaces and the protocols required for higher bandwidth. The Spartan-3E device I/O pins support full functionality for fast, flexible electrical interfaces. The PCI Express slots are 100 MHz compatible. Also there are 18 I/O standards, DDR I/O registers, DCMs.

4.1.2. High Performance DSP Solutions

Spartan-3E FPGAs help you efficiently build DSP solutions that handle. Up to 9.1 billion multiply and accumulates (MACs) per second. There are up to 36, 18x18 embedded multipliers for implementing compact DSP structures such as MAC engines, and adaptive and fully parallel FIR filters. The Block RAM can be used for storing partial products and coefficients.

4.1.3. Lowest Cost Embedded Processing Solutions

The effective fractional cost of incorporating the Micro Blaze (32-bit soft processor) into a Spartan-3E FPGA is very less. The Xilinx Micro Blaze with Spartan-3E FPGA (Figure 2.4) can be used to integrate the entire processing engine, all control functions, and additional supporting logic into a single cost-effective platform. The Embedded Development Kit (EDK) offers a common development environment for Spartan Series FPGAs with Micro Blaze.

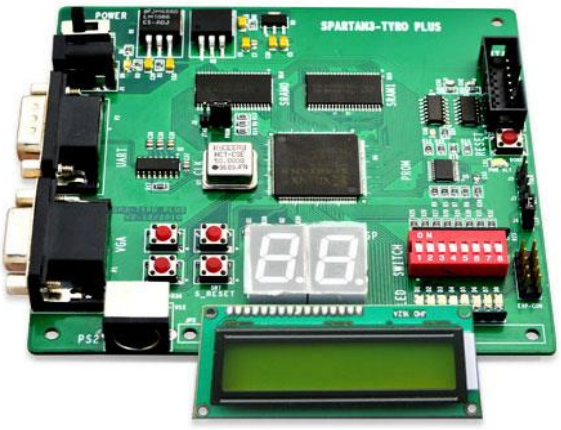


FIG: Spartan 3E Starter Kit

4.2. Development Tools:

The FPGA/FPGA chip is supported with a complete set of software and hardware development tools - Xilinx Embedded Development Kit (EDK) and Xilinx Platform Studio (XPS) tools development software. This tool is used to create a simple processor system. The microprocessors available for use in Xilinx Field Programmable Gate Arrays (FPGAs) with Xilinx EDK software tools can be broken down into two broad categories. There are soft-core microprocessors (Micro Blaze) and the hard-core embedded microprocessor (PowerPC). EDK uses Intellectual-Property Interface (IPIF) library to implement common functionality among various processor peripherals. It is verified, optimized and highly parameter able. It also gives you a set of simplified bus protocol called IP Interconnect (IPIC). Using the IPIF module with parameterization that suits your needs will greatly reduce your design and test effort because you don't have to re-invent the wheel. This is done in EDK with a wizard that walks you through the entire process.

4.3. Algorithm Mapping:

The FPGA implementation is divided into blocks, each block implementing a separate portion of the algorithm. This approach allowed for concurrent development and for testing of individual blocks. The inbuilt finite state machine (FSM) controls each block. In addition, a high-level FSM controls the interaction of the blocks. Each computational block is implemented in C and checked for proper functionality with simulators (ISE Simulator). Conceptually, each pixel in the output image is produced by sliding an N×N window over the

input image and computing an operation according to the input pixels under the window and the chosen window operator. The result is a pixel value that is assigned to centre of the window in the output image.

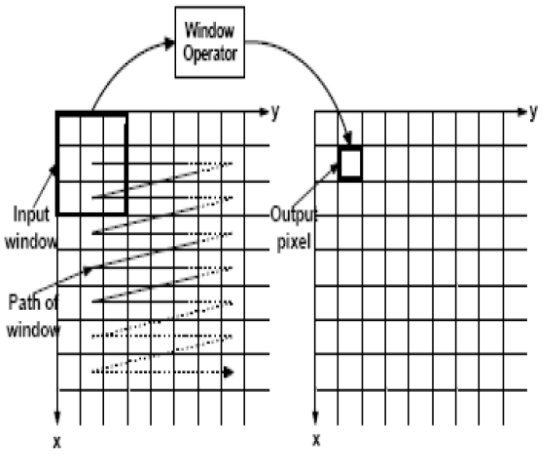


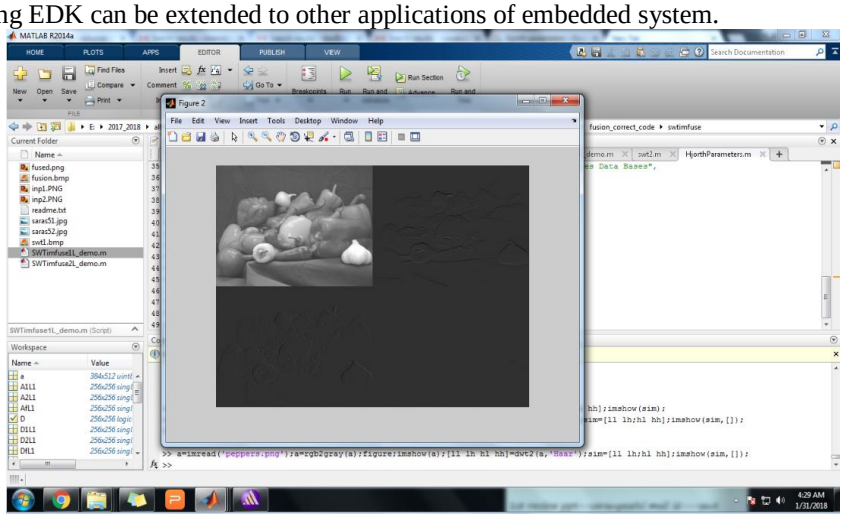
Fig: Mapping The Window Operation

For processing purposes, the straightforward approach is to store the entire input image into a frame buffer, accessing the neighbourhood pixels and applying the function as needed to produce the output image. If processing of the video stream is required N×N pixel values are needed to perform the calculations each time the window is moved and each pixel in the image is read up to N×N times. Memory bandwidth constraints make obtaining all these pixels each clock cycle impossible. Input data from the previous N-1 rows can be cached using a shift register (or circular memory buffer) for when the window is scanned along subsequent lines.

A better option is to replicate the edge pixels of the closest border. Such image padding can be considered as a special case of pipeline priming. When a new frame is received the first line is pre-loaded into the row buffer the required number of times for the given window size.

V. CONCULSION:

This paper presented an approach towards VLSI implementation of the lifting based Discrete Wavelet Transform (DWT) for image compression. The architectures are representative of many design styles and range from highly parallel architectures. Here a DWT-based reconfigurable system is designed using the EDK tool. Hardware architectures of two dimensional (3-D) DWT have been implemented as a coprocessor in an embedded system. In addition, the hardware cost of this architecture is compared for benchmark images. This type of work using EDK can be extended to other applications of embedded system.



5.1. FUTURE SCOPE:

Substantial gaps to compression limits still exist Trend toward algorithms to handle large, multidimensional images Trend to multiple core processors to spur development of new parallel processing paradigms Open question whether quantum information theory and quantum computation will save the day Here we have written

the core processor Micro blaze is designed in system C Language, implemented using Xilinx platform studio and tested in SPARTAN-3 FPGA kit by interfacing a test circuit with the PC using the RS232 cable.

5.2. REFERENCE:

[1] Rakesh Biswas, Siddarth Reddy, Swapna Banerjee, Senior IEEE Member "A High Precision-Low Area Unified Architecture for Loosy and Lossless 3D Multi-Level Discrete Wavelet Transform", TCSTV.2017.2721113, IEEE Transaction on Circuits and System for Video Technology,2017.

[2] P.K. Meher, B.K. Mohanty, and M.M.S. Swamy," Low-Area and Low-Power Reconfigurable Architecture for Convolution-Based 1-D DWT Using 9/7 and 5/3 Filters," in *Proc. 28th International Conference on VLSI Design (VLSID),* pp. 327-332, Jan. 2015.

[3] A. Darji, S. Shukla, S.N. Merchant, and A.N. Chandorkar," Hardware Efficient VLSI Architecture for 3-D Discrete Wavelet Transform," *Proc.27th Int. Conf. on VLSI Design and 13th Int. Conf. on Embedded Systems, Mumbai,* pp. 348 - 352, 5-9 Jan. 2014.