International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

VLSI IMPLEMENTATION OF REAL TIME MODIFIED DECISION BASED MEDIAN FILER USING DATA COMPARATOR

[1]K.Malathi, [2]K.Suresh Babu, [3]Y.Akhila, [4]U.Keerthi, [5]T.Yasaswini

[2]Associate Professor, ECE Dept , RISE Krishna Sai Prakasam Group of Institutions, Valluru, AP

[1,3,4,5]B.Tech final year students, ECE Dept, RISE Krishna Sai Prakasam Group Of Institutions, Valluru, AP

[1]malathikunduru2003@gmail.com , [2]suresh.raja001@gmail.com, [3]akhilayekambaram121@gmail.com, [4]udumulakeerthi3@gmail.com, [5]telagathotiyasaswini2004@gmail.com

ABSTRACT

The VLSI-oriented hardware implementation of filters is essential for real-time applications because they are used to remove various types of noise from images, such as gaussian, random, and salt and pepper noises. However, conventional hardware-based filters are unable to reduce look-up-tables (LUTs), path delays, and power consumption. For this reason, this work focuses on the implementation of a Hybrid Median Filter (HMF) using Data Comparator (DC) logic. First, a data comparator based on multiplexer selection logic is used to identify the high and low values from two numbers. The data comparator is then repeated numerous times for nine pixel combinations, identifying the median value from nine pixels. Comparing the suggested HMF-DC to state-of-the-art methods, the subjective and objective evaluation demonstrates that it performed better in terms of lower noise, hardware metrics such LUTs, delay, and power consumption.

Keywords:

VLSI Design, FPGA Implementation, Median filter, MATLAB, Data Comparator.

1.INTRODUCTION

Electronic circuits in general and memories in particular struggle in harsh conditions, such as space. Radiation, for instance, produces a variety of faults that might impair circuit functionality [1]. Soft errors, which alter the value of one or more memory cells, are a frequent error for SRAM memories [2]. Error correction codes (ECCs) are frequently employed to prevent corruption in the data stored in the memory [3]. To find and fix mistakes, ECCs append parity check bits to every memory word. This calls for a decoder to identify and fix mistakes when reading from the memory and an encoder to calculate those bits when writing to the memory. These components can lower access speed while increasing memory size and power consumption. Codes that can fix a single bit error per word have been employed historically. Specifically, it is usual practice to utilize single error correction–double error detection (SEC–DED) codes that are also capable of detecting double errors [4]. Errors that impact several memory cells have become much more common in recent years.

This is because the memory cells are scaling and are expected to continue growing [5]. SEC-DED codes are challenged by these mistakes, which are referred to as multiple cell upsets (MCUs). One solution to ensure that the MCU errors can be corrected is to interleave the bits

of different logical words so that an MCU affects one bit per word [6]. This is predicated on the finding that the cells that an MCU affects are spatially near to one another [7]. However, interleaving comes at a price because it makes theme memory design more difficult [8]. Another problem in some space applications is that there are a lot of errors, and SEC-DED codes might not be enough when errors mount up over time [9]. Because of these problems, there is now growing interest in using more sophisticated ECCs to safeguard SRAM memories. Numerous codes that can fix double-adjacent or triple-adjacent faults have lately been developed since MCUs impact cells that are close to one another[8],[10]–[12]. Most of the time, these codes don't need extra parity check bits; the others merely need one or two extra bits. Although the decoding complexity rises, it is frequently still possible to implement it with little effect on memory speed. These codes are helpful for situations when the error rate is low, but codes that can fix faults on several independent bits are required when the error rate is high [9]. Because typical decoders are often serial and take multiple clock cycles, research for multibit ECCs has concentrated on lowering the decoding delay.

Nevertheless, the Golay code necessitates a more intricate decoder that requires multiple clock cycles [17]. A compromise solution for Bose–Chaudhuri–Hocquenghem codes has been put forth by Namba et al. [18]. The most frequent mistake patterns are supposed to be decoded in simultaneously, while the others should be processed serially. Specifically, a single clock cycle is used to repair single and double-adjacent faults. This indicates that just a tiny portion of the incorrect words need a complete serial decoding, and that the greatest number of memory accesses can be finished in a single clock cycle. This can make it possible to safeguard SRAM memory using conventional ECCs that don't provide fast parallel decoding. This brief examines the application of the [18] scheme to the (24,12) Golay code. More specifically, an effective parallel decoder that can fix both single- and double-adjacent mistakes is shown. To lower the implementation cost, the decoder takes advantage of the Golay code's characteristics. This produces a decoder that can fix all double-adjacent faults and some triple-adjacent errors, while also being less complicated than a conventional SEC decoder. To demonstrate the advantages of the suggested decoder, it has been mapped to a 65-nm technology and implemented in hardware description language.

2.LITERATURE SURVEY

A very helpful combinational circuit called a comparator is used to determine whether a binary number at one input is more or less than another binary number. An essential comparator can be an XOR gate. There are two categories of Comparators (a). Comparator of Magnitude (b). A comparison of data. While the latter provides the greater and lesser data itself, the former just compares the two binary integers' magnitudes. To show if the first input is greater than the second or vice versa, the magnitude comparator provides two outputs. However, because it compares words X and Y and returns a higher and lower value, respectively, a data comparator is also known as a two cell comparator. High performance, low power consumption, and a compressed, high-quality cost are important factors in practically all hardware comparators. In comparison to current circuits, the effort aims to observe the characteristics of specific comparator circuits that guarantee superior performance. An Adaptive Rank Order filter (AROF) with VLSI implementation was created by Karpagaabirami and Ramamoorthy [1] to

eliminate impulsive noise and pipelining with parallel processing to expedite the filtering process. Compared to the Decision Tree Based De-noising Method (DTBDM), the Decision Rank Order Filter (DROF) has the advantage of requiring less space and having a simpler architecture.The disadvantage of VLSI DTBDM involves too many architectures for detection of noise and reconstruction of noisy pixel.

Ayeesha et al [2] explored the design of high speed and low power comparator since it operates only with 1volt power and less propagation delay and its architecture includes two stage CMOS op-amp circuit. In this work, comparator is designed with cadence tool with 0.18micrometer technology.

Bharat et al [3] introduced a special types of comparators and these circuits are simulated with 1 Volt DC supply voltage in LTspice-IV using PTM 45nm technology. Unlike static and dynamic characteristics of all these comparators are considered and compared and it operates with higher speed and provide more stabilized output compare to 90nm and 180nm.

Vasanth et al [5] developed a parallel architecture and pipelined architecture for modified shear sorting. The method introduced an area efficient data comparator for sorting 9 elements. The basic processing element is area optimized two cell sorter. A group of three two cell sorter form a three cell sorter which in turn uses compare and swap approach for ordering the data sequence.

Keeping chan [6] developed a VLSI algorithms and implementation architectures for a class of nonlinear filters. The class of filters contains all of the functions earlier defined by stack filters, where rank-order and median filters are special cases. The function of a stack filter can be realized in k-step recursive use of one binary processing circuit.

3.EXISTING SYSTEM

3.1 Basic Structure of Decision based multiplexer

The basic structure for the Proposed Decision based multiplexer consists of two inputs and two outputs. The internal structure of the decision based multiplexers consists of a logic circuits that identifies which of the two inputs is greater by generating a signal which is in turn drives two multiplexers which yield a high or a low value depending upon the generated signal.

3.1.1 Conventional bitwise decision based multiplexer

Conventional decision based multiplexer uses the methodology of magnitude comparator and acts like a word comparator. A 1- bit magnitude comparator is used as a processing element for 8-bit comparison where a, b are two 1 bit number and there are two outputs corresponding to a>band.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

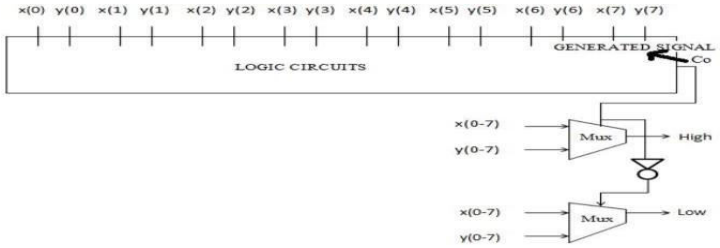


Fig.3.1.1.Conventional bitwise decision based mux block diagram

3.1.2. Carry select logic decision based multiplexer

The carry select logic comparator design was developed for an 8-bit comparator. To implement a comparator an alternative basic operation is a subtraction. Hence we use a rippled borrow output of an 8 bit full subtractor to select the data from the multiplexer.

3.1.3. Borrow look ahead logic decision based multiplexer (blac)

The borrow dependency problem is eliminated using borrow look ahead logic decision based multiplexer.

3.1.4. Decoder based decision based multiplexer

The Decoder-based comparator is used to control the flow of logic i.e data transfer [7]. The proposed logic circuit is simple and effective that compares MSB of first 8-bit with MSB of another 8-bit binary number and provides the output by performing basic XOR gate operation of X and Y respectively.

3.1.5. Multiplexer based decision based multiplexer

A Processing Element (PE) consists of OR gate, AND gate and Multiplexer. The inspiration of the work is derived from Keshab parhi adder using multiplexer [12]. The truth table of full subtractor is shown in Table I. The table does not require a difference as we are interested only in borrow out. **Two's complement based data converter:**

This decision based multiplexers uses twos complement implementation to implement a Decision based multiplexer. The basic twos complement implementation is given by A+B'+1. In this design, B'+1 is implemented using binary to excess one converter instead of a full adder.

3.2. MODIFIED SHEAR SORTING (MSS)

Modified shear sorting [5] is simple and fast algorithm which is used to determine median of list of elements using compare and swap operation. 3.

3.2.1. Parallel architecture for modified shear sorting

The methodology of modified shear sorting is implemented using compare and swap operation that works as a parallel architecture. The different architecture for decision based multiplexer also referred as two cell sorter is used to build a small processing element called three cell sorter. A three cell sorter is a basic unit that compares three elements and gives out Maximum, median and minimum value of three elements.

4.PROPOSED SYSTEM

Unwanted information that lowers visual quality is called noise. [1]. Dust on the lens, electronic noise in the camera, flaws in the image sensor, or errors introduced during the transmission of picture data across a communication channel can all generate a noisy image. Removing noise from a digital image without changing its qualities is the goal of image processing. The main blog of the image processing system is the image filter. When image data is sent over an unsecure communication connection, an impulsive noise may be introduced. [2]. It results in dark or black spots or tiny dots on an image. The most frequently reported noise in digital photos is impulse noise, which is evenly distributed. Additionally, impulse noise can be separated into two categories. The first is salt and pepper noise, a kind of impulse noise with noisy pixel intensities that, for grayscale images, range from 0 (minimum) to 255 (highest). Over the photos, it appears as haphazardly dispersed black or white dots [3]. The acquired image must go through an image pre-processing step known as a filter in order to eliminate these sounds [4]. There are two types of filtering operations: frequency domain and spatial domain. In real-time systems, filters are typically developed using MATLAB and OCTAVE software [5]. The fact that software implementation gives a lower processing speed than hardware implementation is well known [6]. With the advancement of VLSI technology, hardware implementation has emerged as a superior alternative. ASICs, DSPs, and embedded microprocessors make up the majority of these items [7]. Achieving the low force plan of any VLSI circuit is a challenging task. For low force applications, there are different levels of development in VLSI configuration measures. Power has been the primary challenge for portable devices that run on batteries [8]. The System-on-Chip (SoC) needs less power as it develops with more power transistors. Reducing power consumption in highly integrated SoCs eliminates the issue of heating. It lowers the price of pricey cooling systems and packing [9]. The aforementioned problems of power and cost reduction are addressed in this study by proposing a VLSI architecture for noise reduction in various imaging applications. This work primarily focuses on Verilog-based coding techniques with an FPGA prototype in order to attain low resources [10]. The MATLAB environment is then used to measure the subjective and objective image statistics. The following are this work's primary contributions:

- Implementation of data comparator for identifying the high, low values using multiplexer selection logic.
- Implementation of multi-level network for selection of median value from nine pixels in a window.
- Implementation of HMF-DC for removal of different types of noises from image using hybrid switching of data blocks.

The article's remaining sections are arranged as follows: The literature review is covered in section 2, the suggested HMF-DC implementation is covered in section 3, the findings are analyzed and performance comparisons are made in section 4, and the study is concluded with potential future directions in section 5. It is difficult to remove noise without destroying details in the image because it is signal-subordinate. A variety of error types, including Gaussian, drive, dot, and Rician commotions, affect the image. Information on the type of fault in the

original image plays a significant role in the image denoising process. Either multiplicative or additive substance can be assigned to the picture error. Where f is the amplitude at any spatial coordinate x and y, the image is a 2-D function $f(x, y)$ of light intensities.

A light beam strikes an item, and the reflected light reaches the eyes. People are able to see the thing. Pixels are the image's smallest component.

$$F(x, y) = I(x, y).\ R(x, y) \qquad (1)$$

Here, $I\ (x, y)$ represents the intensity of incident light on the object, $R\ (x, y)$ represents the intensity of reflected light from the object, and $F(x, y)$ represents the intensity of the resulting image. The technique of denoising an image that has been warped by prior knowledge of the degradation model is known as image restoration. After determining the deterioration model, the intended imagery can be recovered by using the inverse technique. Conventional picture enhancement methods are not the same as image restoration. It's a subjective procedure that yields better outcomes for an observer both with and without the degradation model. Figure 1 illustrates the process of image degradation. Convolution between $f\ (x, y)$ and model function $(h(x, y))$ is used to create an image deterioration model in the spatial domain.

$$F(x, y) = h(x, y) * f\ (x, y) + \eta\ (x, y) \qquad (2)$$

In this case, the speckle noise is represented by $\eta\ (x, y)$. Additionally, the Fourier transform is used to create a degradation model in the frequency domain in the manner described below:

$$F\ (u, v) = h\ (u, v) * f\ (u, v) +\ \eta(u, v) \qquad (3)$$

Here, u, v represents the frequency domain coefficients.

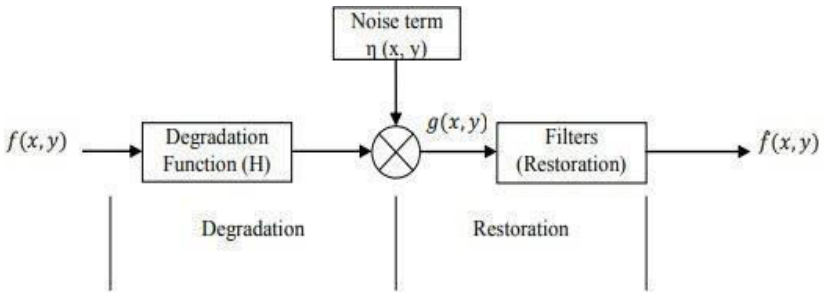


Fig.4. image degradation model

The HMF-DC is a digital non-linear approach used to remove noise, similar to that of the medium filter. However, it usually performs better than the mean filter by preserving important information in the image. This filter class is a member of the edge-preserving filter class. These filters preserve the intricacies while smoothing out the data. Simply put, the median is the mean of all the pixel values within the region. The median is half larger and half smaller in the neighborhood, but it doesn't match the mean (or average). A "centre indication" that is more powerful than the average is the median. Similar to the medium, the HMF-DC considers each pixel in the image and analyzes its nearby pixels to see if they are representative of their environment.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

It is especially effective in eliminating impulsive noise compared to the standard filter. Since there is no discernible difference between the noise and the tiny details, the HMF-DC removes both. The median value will be reduced and eliminated by anything that is relatively little in relation to the region size. Stated differently, the HMF-DC is able to distinguish between noise and fine details.

4.1.Data comparator

Figure depicts the block diagram of data comparator, which is used to execute the selection of highest and lowest values from the given two input data. Additionally, inputs A and B and outputs H and L are present in the data comparator block. In step one, the A < B condition is first checked; if it is met, the multiplexer's selection line becomes one; if not, it becomes zero. Step 2: The 2to1 multiplexer receives inputs A and B as Data-input-0 and Data-input-1, respectively. The selection line becomes one if A is less than B, and the multiplexer uses selection switching to produce H as input-B. Selection switching causes the selection line to become zero and the multiplexer to produce H as input-A if A is greater than B.

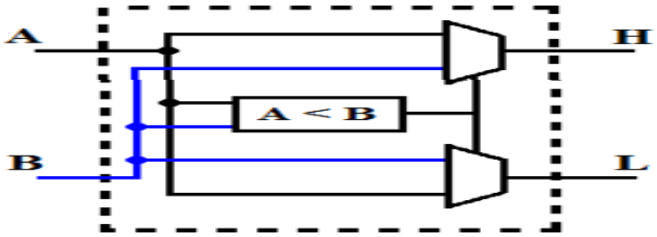


Fig.4.1. Block diagram of data comparator.

Step 3: The 2 to1 multiplexer receives inputs B and A as Data-input-0 and Data-input-1, respectively. The selection line becomes one if A is less than B, and the multiplexer uses selection switching to produce L as input-A. Selection switching causes the selection line to become zero and the multiplexer to produce L as input-B if A is greater than B.

4.2.Hardware architecture of HMF

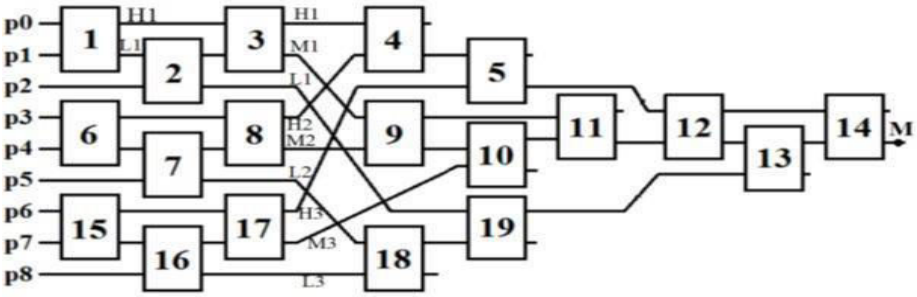


Fig .4.2.(a) Hardware architecture of HMF-DC.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

The hardware architecture of HMF-DC, which includes fourteen hardware resource blocks, is depicted in Figure 3. The median value, M, is produced by applying inputs P0, P1, P2, P3, P4, P5, P6, P7, and P8 to HMF-DC. In this case, DC-1, DC-2, and DC-3 are combined together and pick the median (M1), low (L1), and high (H1) values. Likewise, DC-6, DC-7, DC-8, and DC-15, DC-16, and DC-17 generate high, low, and median values. Additionally, the lowest value from the H1, H2, and H3 results is chosen using DC-4. Additionally, the highest value from the L1, L2, and L3 results is chosen using DC-18. DC-9, DC-10, and DC-11 are also grouped together and pick high, low, and median values. The procedure will proceed in this manner, producing the median value (M) from the DC-14 low result.

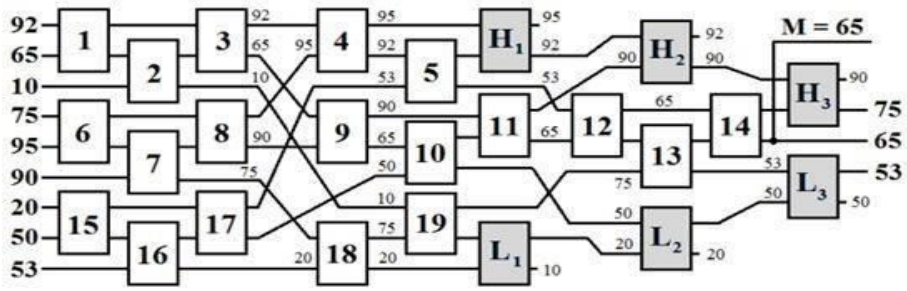


Fig 4.2.(b) Example of HMF-DC.

A numerical example that may help clarify the operation of the HMF-DC system is shown in Figure 4. The two non-median outputs that are closest to the median value in this instance define the median value. This diagram illustrates how the four highest pixel values (95, 92, 90, and 75) are sorted using the H1, H2, and H3 blocks, with 75 serving as the upper range. The four lowest values (10, 20, and 50) are simultaneously sorted using the three lower ranges (L1, L2, and L3), with 53 being the bottom range.

5.RESULTS AND DISCUSSION

All of the HMF-DC designs were made using Xilinx ISE software. Simulation and synthesis are the two output types that this software program produces. A detailed analysis of the HMF-DC architecture in terms of input and output byte level combinations is given by the simulation results. The decoding process is roughly accomplished by applying multiple input combinations and keeping an eye on different results from a simulated investigation of the accuracy of the encoding. The results of the synthesis will be used to determine the area in respect to the LUT count. Furthermore, a power summary will be created using the static and dynamic power usage, and a time summary pertaining to different path delays will be acquired.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

Furthermore, the subjective performance of HMF-DC is assessed using MatlabR2020a program.

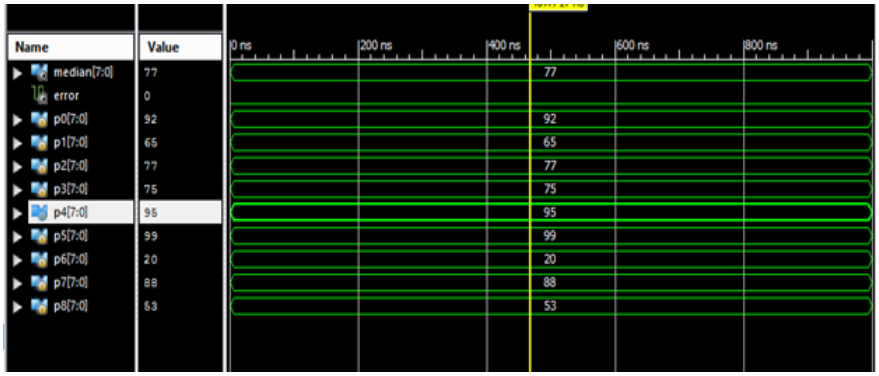


Fig .5.1. Simulation outcome of HMF-DC.

Figure presents the simulation outcome of HMF- DC. Here, P0, P1, P2, P3, P4, P5, P6, P7, P8 are the inputs to HMF-DC and median is the output value.

Utilization Post-Synthesis | **Post-Implementation**

Graph | **Table**

Resource	Utilization	Available	Utilization %
LUT	399	47200	0.85
IO	81	285	28.42

Fig.5.2 . Design summary.

Unconstrained Paths - NONE - NONE - Hold

General Information
Timer Settings
Design Timing Summary
Check Timing (0)
Intra-Clock Paths
Inter-Clock Paths
Other Path Groups
User Ignored Paths
Unconstrained Paths
NONE to NONE
Hold (9)

Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock	Destination Clo
Path 1	∞	8	6	10	p1[5]	error	3.538	1.724	1.815	-∞	input port clock	
Path 2	∞	5	4	10	p1[0]	median[0]	3.609	1.562	2.047	-∞	input port clock	
Path 3	∞	5	4	10	p2[1]	median[1]	3.650	1.544	2.106	-∞	input port clock	
Path 4	∞	5	4	10	p1[3]	median[3]	3.725	1.504	2.221	-∞	input port clock	
Path 5	∞	5	4	10	p0[5]	median[5]	3.771	1.594	2.177	-∞	input port clock	
Path 6	∞	5	4	10	p0[7]	median[7]	3.793	1.580	2.214	-∞	input port clock	
Path 7	∞	5	4	10	p1[2]	median[2]	3.912	1.564	2.347	-∞	input port clock	
Path 8	∞	5	4	10	p1[6]	median[6]	4.021	1.640	2.381	-∞	input port clock	
Path 9	∞	5	4	10	p1[4]	median[4]	4.208	1.593	2.615	-∞	input port clock	

Fig.5.3 . Time summary.

The design (area) summary of the suggested approach is displayed in Figure 6. In this case, the suggested approach makes use of the low area in terms of slice LUTs, or 437 of the 303600 available. The suggested method's time summary is displayed in Figure 7.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
IJARST
ISSN: 2457-0362

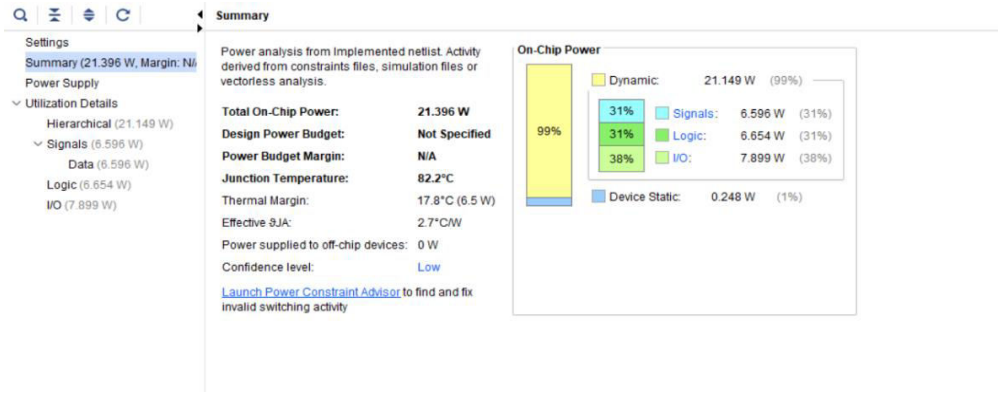


Fig.5.4: Power summary.

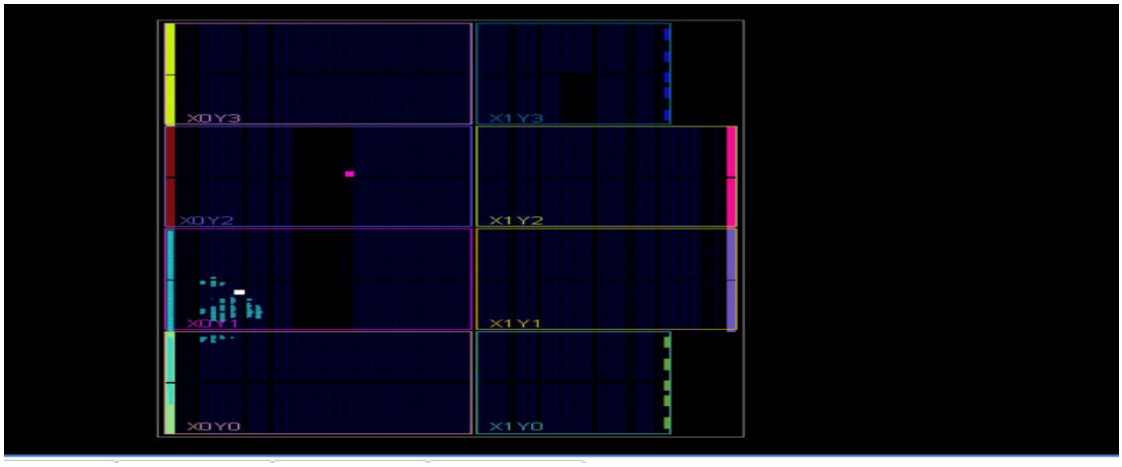
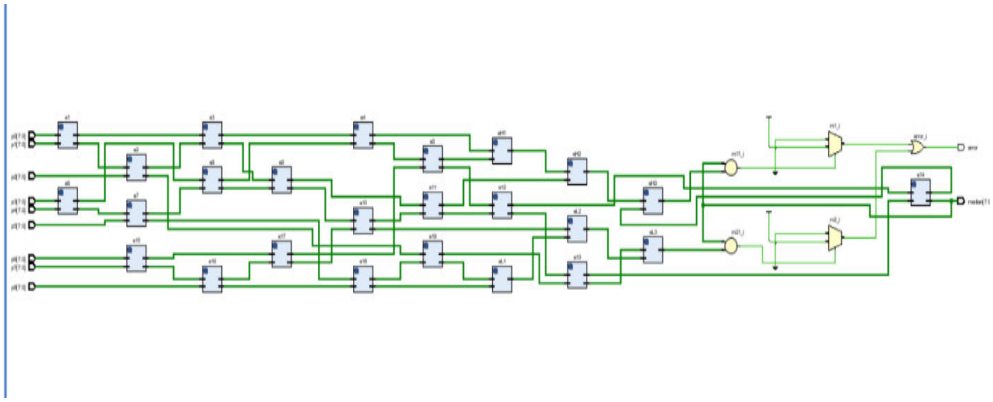


Fig.5.5: Internal Simulation Result



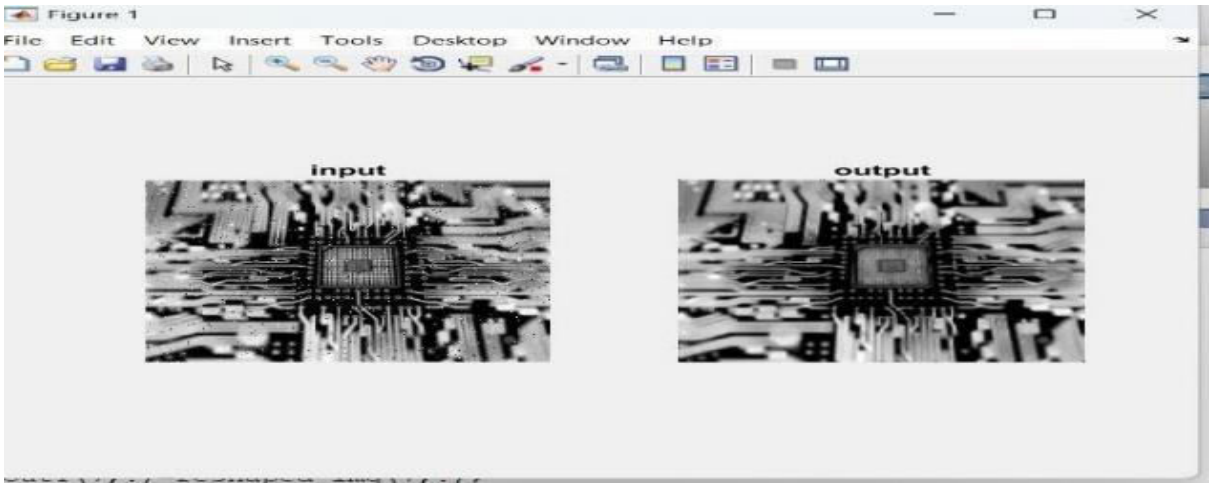
International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

Fig.5.6: Schematic Diagram of data comparators

Fig.5.6: Image output

The filtering performance of several techniques, including SMF [18], DMF [21], AMF [25], and the suggested HMF-DC, is displayed in the figure. Here, the results of the DMF [21] approach contain low level noises, whereas the results of the SMF [18] and AMF [25] methods still contain greater noises. However, the output of the suggested HMF-DC approach resembles the original image. The performance evaluation of the suggested HMF-DC approach is contrasted in Table 1. Comparing the suggested HMF-DC to more traditional methods like SMF [18], DMF [21], and AMF [25], the former produced better (lower) hardware performance in terms of LUTs, time-delay, and power consumption.

6.CONCLUSION



The main focus of this work is the creation of a hybrid median filter using data comparator logic. The first step is to identify which of two integers has high and low values using a data comparator based on multiplexer selection logic. Following that, the data comparator is run several times for each of the nine possible pixel combinations, calculating the median value for each of those nine values. In terms of reduced noise, latency, and power consumption, the proposed HMF-DC outperformed the state-of-the-art methods, according to both the subjective and objective evaluations

REFERENCES

[1]. Goel, Anish, M. N. S. Swamy, and M. Omair Ahmad. "Design of a 2D median filter with a high throughput FPGA implementation." IEEE's 62nd Midwest Symposium on Circuits and Systems (MWSCAS) in 2019. IEEE, 2019.

[2]. Bongsoon Kang, Gi-Dong Lee, Ngo, and Dat. "A 4K-Capable FPGA Implementation of Single Image Haze Removal Using Hazy Particle Maps." 3443 in Applied Sciences 9.17 (2019).

[3]. Rahnama, O., Di Stefano, L., Joy, T., Cavallari, T., Golodetz, S., Tonioni, A.,... & Torr, P. H. (2019). fpga-cpu hybrid SOC for real-time, very accurate dense depth on a power budget. IEEE Circuits and Systems Transactions II: Express Briefs, 66(5), 773-777.

[4]. Kamaraju, M., and N. Sambamurthy (2020). Power-efficient median filtering based on hybrid sorting. Digital Signals and Smart Systems International Journal, 4(1-3), 80-86.

[5]. Súcar, L., and J. Arnal (2019). Hybrid filter based on fuzzy approaches for mixed noise reduction in color images. Applied Sciences, 10(1), 243.

[6]. Ahmad, Rais, and Som Pal Gangwar. "Rain Streaks Elimination Using Hybrid Median Filter and Contrast Stretching." advances in VLSI, communication, and signal processing. Springer, Singapore, 2021. 157-

[7]. Rajasekaran, M. P., Rini, C., and Perumal, B. (2019). Using a local mean-based hybrid median filter, Rician noise in orthopedic knee MR images is eliminated. Proceedings of the Second International Conference on Communication Technology and Data Engineering, pp. 693-702. Singapore: Springer.

[8]. Banerjee, S., Mehra, R., Misra, A., & Sinha Chaudhuri, S. (2021). An in-depth analysis of the median filter, hybrid median filter, and suggested variants. Developments in Information Processing and Smart Communication Technologies, 393–406.

[9]. Ibraheem, B. A., Mustafa, Z. A., Elsharif, R. I., & Abass, S. K. (2018). Speckle-reducing anisotropic diffusion and hybrid median are used in this wavelet decomposition-based speckle reduction technique for ultrasound pictures. Clinical Engineering Journal, 43(4), 163-170..

[10]. Deepa, D., and Kannan, L. M. (2021). Finite impulse response (FIR) filter design for biological imaging applications using low power very large scale integration (VLSI). 96(5) DYNA-Ingeniería e Industria.

[11]. In October 2020, Yang, C. H., and Lin, C. H. Denoising filter and local binary pattern feature extraction architectures combined. (pg. 152-153) IEEE 9th Global Conference on Consumer Electronics (GCCE) 2020. IEEE.