

# Novel Highspeed Architecture for Median Filter

G. Isaac Jerose<sup>1</sup>, R. Senthamizh Selvi<sup>2</sup>

<sup>1</sup>PG Scholar, Department of Electronics and Communication Engineering, Easwari Engineering College, Chennai, Tamil Nadu, India  
isaacjeroseg@yahoo.com

<sup>2</sup>Asst.Prof.Sl.Gr, Department of Electronics and Communication Engineering, Easwari Engineering College, Chennai, Tamil Nadu, India  
sachin92k4@yahoo.com

**Abstract:** *This paper deals with developing a efficient VLSI architecture for median filter to remove impulse noise. Median filter is one of the important non-linear filters, used in speech and image processing applications. In this paper, four new low complexity and high speed architectures are proposed for 5x5 median filters. The main idea of this paper is to employ various techniques Such as, fine grain pipelining, pipelining and unfolding to increase speed of the architecture. The proposed 5x5 architecture is developed using five cell, three cell and two cell sorters which gives median value with low complexity and high speed. First architecture is developed using fine grain pipelining technique in which registers are introduced in the comparator circuit itself, so that delay is reduced from 164 ns to 24ns. Second architecture is developed using pipelining technique in which registers were inserted across the feed forward cutest, so that delay is reduced from 24 ns to 14ns. Third architecture is developed using unfolding technique, it performs the same operation J number of times where J is the unfolding factor. It performs the operation two times (J=2) with delay 23ns. The proposed high speed and low complexity architecture will retain original image quality and sharpness.*

**Keywords:** Median Filter, Pipelining, Fine Grain pipelining, carry logic sorter, unfolding.

## 1. Introduction

In recent years there have been many improvements in nonlinear filters. Non linear filtering algorithm, used in image processing because images do not produce linear characteristics. In previous models, the hypotheses of Gaussianity and stationarity were used to produce linear signals; however, these filters were not able of removing all kind of noise without affecting the signal. For instance, impulsive noise can be removed from an image by using a linear filter; however, edges and small details in the images are blurred in the filtering process.

Channels are used to transmit the images and signals. During this transmission, signals are contaminated by noise due to faulty communication. Noises are broadly classified into additive noise and multiplicative noise. There are various filtering techniques, used to remove the noise from the signal.

Initially linear filters which were used in image processing problems cannot deal with the nonlinearities of the image and the non-linearity's of human vision model cannot be taken into account, because classical linear digital filters have low pass characteristics, they have a tendency to blur the edges, do not eliminate non additive noise and destroy edges and small details of an image.

Multiplicative noise occurs during digital image restoration. Linear filtering technique is not useful in these cases because this does not satisfy the superposition property. Nonlinear filtering techniques are used to overcome the limitations of linear filters. Design and implementation of the nonlinear filters involves without any complex optimization techniques.

A nonlinear filter is a filter whose output is not linear with its input. According to terminology the filtering problem

may refer to the time domain (state space) representation of the signal or to the frequency domain representation of the signal.

Nonlinear filters place and eliminate data that is known as noise. The nonlinear algorithm is checked at each data point whether it is noise or valid signal. If the point is noise, the data point is eliminated and replaced by an estimated data point based on surrounding data points, and parts of the data that are not considered as noise are not modified at all. Linear filters, lack in such a decision capability and therefore modify all data

### 1.1. Impulse noise

Faulty pixels in camera sensors, faulty memory locations in hardware, or faulty transmission channel causes impulse noise occur. Impulse noises are classified into two type's salt and pepper noise, and the random valued noise. Due to salt and pepper noise an image will have dark pixels in bright regions and bright pixels in dark regions Salt and pepper noise caused by analog-to-digital converter errors, bit errors in transmission, etc. Dark frame subtraction and interpolating around dark/bright pixels are used to eliminate the salt and pepper noise. There are many works on the restoration of images corrupted by impulse noise. Impulse noise can be removed by median filter because of its noise removing capability and computational efficiency.

### 1.2. Median filter

A median filter is one of the nonlinear digital filter which is able to maintain sharp signal changes and eliminate impulse noise. While linear filter is unable to remove this type of noise without affecting unique characteristics of the signal, median filters have significant advantages over linear filters for this particular type of noise. Therefore median filter is mostly used in digital signal and image

processing applications. While retaining sharp sustained edges in signals, Standard median filters are used to reduce impulse noise.

The window contains odd number of elements, the middle element (median) which is found by sorting the elements in the window. The window is moved along the entire pixel element in the image, if the center pixel is noisy, which is replaced by the median value of the window to reconstruct the original image.

The 5x5 window has center pixel as reference shown in figure 1. The center pixel is checked whether it is a noise or not. The median value found is checked whether it is a noise. If median value is not a noise then center pixel is replaced by median value. This process is repeated with all pixels in the input as reference pixels. The output image obtained will be noise free and less blurred and it is maintain edge preserving property also.

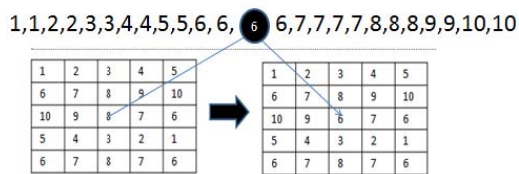


Figure 1: Sorting techniques

## 2. Existing Architecture

The existing architecture takes noisy pixels as input from the noisy image nine pixels are taken by 3x3 windows is shown fig 2. The architecture follows the shear sorting algorithm.

The shear sorting follows a three level sorting:

- First stage:* the row elements of the window are sorted with lowest to highest.
- Second stage:* after sorting row, column elements are sorted.
- Third stage:* following the two steps finally diagonal elements are sorted.

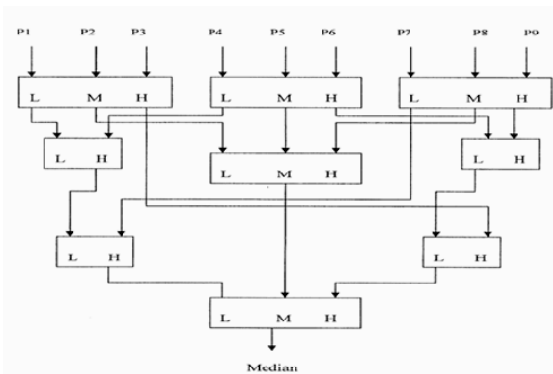


Figure 2: Existing Architecture

## 3. Proposed Architecture

The proposed architecture produces median value from twenty five pixel values. Two input comparator is the essential for all the three cell, five cell sorters.. In the two

input comparator lower value existing on the left, the higher value existing on the right. It consists of one half subtractor, seven full subtractor and two multiplexers. Comparator performs subtraction and swap operation. . The carry of the subtract function is generated using one half sub tractor and seven full subtractors. The borrow of the final subtractor is taken as selection line of the multiplexers. after finding the selection line value, both the input of the comparator are swapped to produce the lowest value at multiplexer 1 and highest value at multiplexer 2. The architecture of the comparator is shown in figure.3.

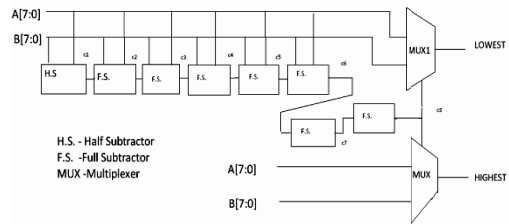


Figure 3: Architecture of Comparator

In the proposed 5x5 window architecture shown in figure 4. The first stage performs the row sorting, the output from first stage given to the input of second stage, which perform the column sorting. Only nine pixels are taken from second stage, out of nine, six pixels are given to two three cell sorter of the third stage. In fourth stage, low value of first three cell sorter, one pixel from second stage is given to two input comparator, middle value of both the three cell sorter and one pixel from second stage is given to input of three cell sorter. Highest value of second three cell sorter, one pixel from second stage is given to input of another two input comparator. In fifth stage, high value fourth stage first comparator, low value of the second three cell sorter in third stage is given to input of first two input comparator, low value fourth stage second comparator, high value of the first three cell sorter in third stage is given to input of first two input comparator. At final stage, high value of the first two input comparator of previous stage, low value of the second two input comparator of previous stage, middle value of the three cell sorter of the fourth stage is given to the inputs of the three cell sorter to find median value.

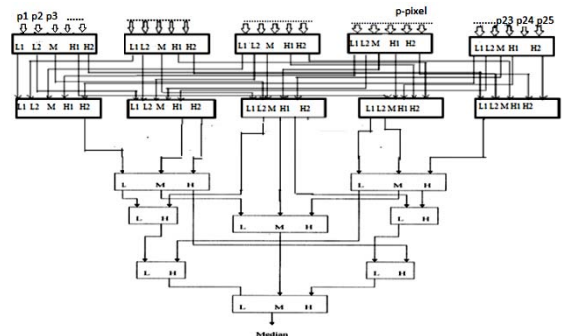


Figure 4: Proposed Architecture

### 3.1. Fine grain pipelining

Fine grain pipelining technique leads to reduction in the critical path and increases the clock or sample period of the architecture. The speed of the architecture is limited by longest path between any two registers or between an input and a register or between a register and an output or between input and the output. This longest path or the critical path can be reduced by suitably placing the pipelining registers in the architecture. In fine grain pipelining technique registers are introduced in the comparator circuit itself, in which nodes of the comparator is split into various sub nodes, registers are placed between them. So the computation time was greatly reduced. By this technique, maximum combinational path delay of the single comparator is reduced from 11.5 ns to 5.7 ns, thereby over all architecture, maximum combinational path delay is reduced from 164 ns to 24ns. The fine grain pipelining architecture is illustrated in figure 5.

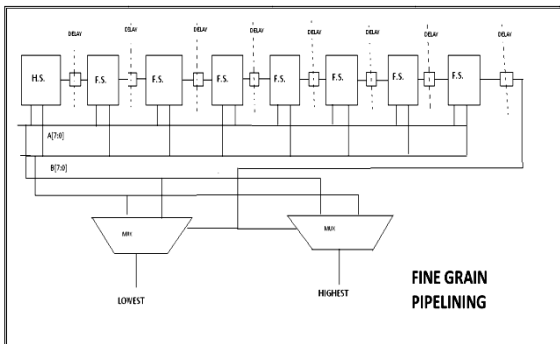


Figure 5: Fine grain pipelining

### 3.2. Pipelining

Pipelining is a transformation technique that leads to reduction in the critical path and increases the clock or sample period of the architecture. The speed of the architecture is limited by longest path between any two registers or between an input and a register or between a register and an output or between input and the output. This longest path or the critical path can be reduced by suitably placing the pipelining registers in the architecture.

Critical path is the longest path among all the paths with zero delay. These pipelining registers are placed across the feed forward cut set of the graph. In the feed forward cutset, data is moved in the forward direction. By this architecture, maximum combinational path delay is reduced from 24 ns to 14.5ns. The pipelining architecture is illustrated in figure 6.

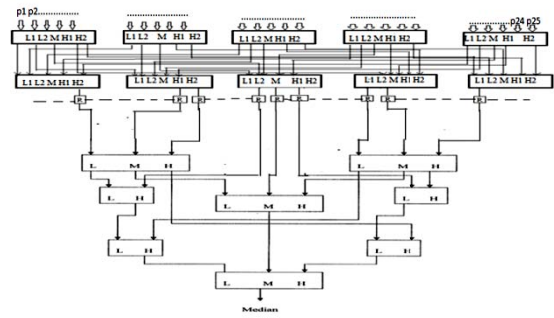


Figure 6: Pipelining Architecture

### 3.3. Unfolding

Unfolding is a transformation technique that can be applied to DSP architecture to create a new architecture describing more than one iteration of the original architecture. Unfolding a DSP program by the unfolding factor J creates a new program that describes J consecutive iterations of the original program.

### 3.4 Unfolding Procedure

$$U_i \rightarrow V_{(i+W)\%J}; W_n = \lfloor (i+W)/J \rfloor;$$

Unfolding Factor J=2

The data flow graph of carry logic sorter is considered for unfolding (J=2). The fig.7 shows the data flow graph of 5x5 window median filters.

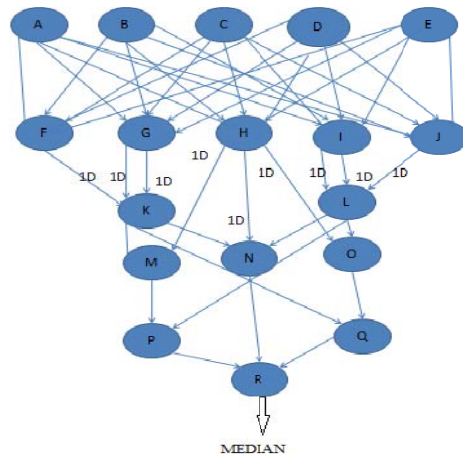


Figure 7: DFG of proposed architecture.

### Calculation:

$$A_0 \rightarrow F_{(0+0)\%2}; A_0 \rightarrow F_0; W_n1 = \lfloor (0+0)/2 \rfloor = 0;$$

$$A_1 \rightarrow F_{(0+1)\%2}; A_1 \rightarrow F_1; W_n2 = \lfloor (0+1)/2 \rfloor = 0;$$

$$A_0 \rightarrow G_{(0+0)\%2}; A_0 \rightarrow G_0; W_n3 = \lfloor (0+0)/2 \rfloor = 0;$$

$$A_1 \rightarrow G_{(0+1)\%2}; A_1 \rightarrow G_1; W_n4 = \lfloor (0+1)/2 \rfloor = 0;$$

$$A_0 \rightarrow H_{(0+0)\%2}; A_0 \rightarrow H_0; W_n5 = \lfloor (0+0)/2 \rfloor = 0;$$

$$A_1 \rightarrow H_{(0+1)\%2}; A_1 \rightarrow H_1; W_n6 = \lfloor (0+1)/2 \rfloor = 0;$$

The calculations are performed and the results are obtained, the unfolded structure of carry logic sorter is designed. Earlier the carry logic sorter finds single median value for the given twenty five input values. Now the unfolded structure will produce two median value outputs for a set of two twenty five input values for the same computation time.

This unfolded structure is very much useful for image processing application, where instead of considering single 5x5 window to remove impulse noise two window of size 5x5 can be taken and image can be processed. The unfolding architecture of the 5x5 window is shown in figure.8.

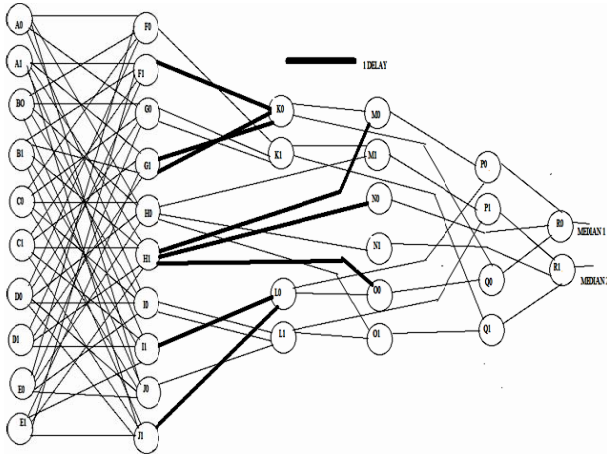


Figure 8: Unfolding Architecture

4. Algorithm

In the proposed algorithm, the 5x5 window is selected for processing pixel and sorting techniques are applied. First the row elements are sorted then column elements are sorted and then diagonal elements are proposed.

5X5 window

$$\begin{bmatrix} 127 & 27 & 25 & 0 & 45 \\ 47 & 87 & 234 & 56 & 34 \\ 29 & 76 & 255 & 64 & 6 \\ 71 & 89 & 34 & 24 & 43 \\ 28 & 96 & 255 & 0 & 0 \end{bmatrix}$$

Step1: The row elements in the 5x5 window are sorted.

$$\begin{bmatrix} 0 & 25 & 27 & 45 & 127 \\ 34 & 47 & 56 & 87 & 234 \\ 6 & 29 & 64 & 76 & 255 \\ 24 & 34 & 43 & 71 & 89 \\ 0 & 0 & 28 & 96 & 255 \end{bmatrix}$$

Step2: The column elements are sorted.

$$\begin{bmatrix} 0 & 0 & 27 & 45 & 89 \\ 0 & 25 & 28 & 71 & 127 \\ 6 & 29 & 43 & 76 & 255 \\ 24 & 34 & 56 & 87 & 234 \\ 34 & 47 & 64 & 96 & 255 \end{bmatrix}$$

Step3: The diagonal elements are sorted.

Diagonal sorting

$$\begin{bmatrix} 0 & 0 & 27 & 45 & 89 \\ 0 & 25 & 28 & 71 & 127 \\ 6 & 29 & 43 & 76 & 255 \\ 24 & 34 & 56 & 87 & 234 \\ 34 & 47 & 64 & 96 & 255 \end{bmatrix}$$

Step4: Sorting the above diagonal values

$$34 \ 34 \ 47 \quad | \quad 28 \ 43 \ 56 \quad | \quad 45 \ 71 \ 89$$

Step5: Sorting low value of first two, median value of all three and high value of last two sorters

$$28 \ 34 \quad | \quad 34 \ 43 \ 71 \quad | \quad 56 \ 89$$

Step6: Sorting high value of first sorter with low value of third sorter and sorting low value of third sorter with high value of first sorter

$$34 \ 45 \quad | \quad 34 \ 43 \ 71 \quad | \quad 47 \ 56$$

Step7: Sorting high value from first, median value from second and high value from third sorters

$$43 \ 45 \ 47$$

Middle value of the sorter 45 is median value

If the center pixel of the window is a noise pixel then it can be replaced by median value (45).

5. Comparison Table

Table 1: Comparison

	Existing architecture (3x3 window)	Proposed architecture (5x5 window)	Fine grain pipelining architecture	Pipelining architecture	Unfolding architecture (for 50 pixels)
Max.combinational path delay in ns	67.01	164.148	24.69	14.5	23.43
Max .frequency in Mhz	-----	-----	48.82	74.35	51.92
Area (no. of. Slices in percentage)	21	120	134	141	138

## 6. Conclusion

In this paper, four architecture were developed using Fine grain pipelining, Pipelining and Unfolding techniques. All the three architecture are comparatively high in speed. The architecture with the combination of fine grain pipelining and pipelining involves a delay of 14.55 ns and the frequency in 74.35 MHz. The Unfolded architecture with  $J=2$  produces two median value with the delay of 23.43 ns. The simulation and synthesis of the architecture has been carried out in Xilinx ISE 13.2 Vertex 6 family for the XC6VLX75T device.

We can state that the developed Pipelined architecture is very useful where the speed is major concern like Remote sensing, Tele medical image processing, Machine Vision and Meteorology, where speed is major concern. They leads to high speed architecture for noise removal.

## References

- [1] M. Govardha, Mrs. Senthamizh Selvi, Dr. K. S. Srinivasan” VLSI Architecture for Nonlinear Filter to Remove Impulse Noise.” International Journal of Environmental Engineering and Management ISSN 2231-1319 Volume 3, Number 1 (2012), pp.119-123)
- [2] K. S. Srinivasan And D. Ebenezer-A New Fast And Efficient Decision-Based Algorithm For Removal Of High-Density Impulse Noise IEEE - Signal Processing Letters, Vol.14, No. 3, March 2007.
- [3] Takeaki Matsubara, Vasily G. Moshnyaga and Koji – Hashimoto-”A Low Complexity Noise removal technique and its hardware implementation”. 978-1-4244-6890-4-2010 IEEE.
- [4] Dargana Prokin and Milan Prokin – “ Low hardware Complexity pipelined Rank Filter ”- IEEE Transactions on circuits and systems- II; Express Briefs, vol.51, No.6, June-2010.
- [5] Pei-Yin Chen, Member, IEEE, Chih-Yuan Lien, Student Member, IEEE, And Hsu-Ming Chuang-”A Low-Cost VLSI Implementation For Efficient Removal Of Impulse Noise”- IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol. 18, No. 3, March 2010.
- [6] Chih-Yuan Lien Pei-Yin Chen, Li-Yuan Chang, Yi-Ming Lin And Po-Kai Chang-”An Efficient Denoising Chip For The Removal Of Impulse Noise” - 978-1-4244-5309-2/10/\$26.00 © 2010 IEEE.
- [7] V. V. Ravi Teja<sup>1</sup>, K. C. Ray, I. Chakrabarti<sup>2</sup>, A. S. Dhar<sup>3</sup> ,-”High Throughput VLSI Architecture For One Dimensional Median Filter”- IEEE -International Conference On Signal Processing, Communications And Networking Madras Institute Of Technology, Anna University Chennai India, Jan 4-6, 2008. Pp339-344

## Author Profile

**G. Isaac Jeroose** received a bachelor’s degree (Francis Xavier Engineering College) in Electronics and Communication Engineering from Anna University of Technology, Tirunelveli, India in 2011 and doing Master’s degree in VLSI design at Easwari Engineering College, Chennai, India.

**R. Sentamizhselvi** received bachelor’s degree from Barathidasan University, Trichy. Master degree in VLSI design from Anna University, Chennai and Pursuing Ph.D in Anna University (CEG), Chennai and having more than seventeen years of teaching experience in Engineering filed.