JES
Journal of Engineering Sciences

Vol 13, Issue 01, Jan /2022
ISSN NO:0377-9254

An Energy Efficient High Throughput Median Filter Using Clock Gating

NEELI SRAVANTHI, PG SCHOLAR, Dept ECE, MATURI VENKATA SUBBA RAO ENGINEERING COLLEGE.

Dr. SVR. MANIMALA, ASSOCIATE PROFESSOR, Dept ECE, MATURI VENKATA SUBBA RAO ENGINEERING COLLEGE.

ABSTRACT

This paper presents a hardware design for an energy efficient, high-speed, one-dimensional median filter. The median filter is a non-linear digital filtering technique, often used to remove noise from an image or signal. Such noise reduction is a typical pre-processing step to improve the results of later processing (for example, edge detection on an image). Median filtering is very widely used in digital image processing because, under certain conditions, it preserves edges while removing noise (but see the discussion below), also having applications in signal processing. Existing architectures focus on operating speeds, thus resulting in redundant power dissipation. This paper presents an algorithm and mathematical model for controlling the clock signals attached to circuit by analyzing the behavior of the filter, which immobilizes the data in registers and reduces not only signal transitions but also switching activities, thereby reducing the total dynamic power consumption. Furthermore, the proposed architecture provides high-speed computation. A median result can be produced in each clock cycle, and the maximum operating frequency performance is nearly independent of the filter size. The proposed architecture uses 90-nm processtechnology and experimental results show that the proposed method is more energy efficient than existing designs. The power consumption is reduced by25% on average.

Index Terms—Low-Power, hardware, median filter, VLSI

INTRODUCTION

The function of a median filter is to calculate the middle-ranked value among all the data in a sample window. For a filter with window size M, the median is the value in position ⌈1⌉ among the sorted M data when M is an odd number, or the average of (M/2)th and (M/2+1)th data when M is even. Median filter is used extensively in digital signal processing and image processing to filter out noise. Smoothing of signals affected by impulse noise or salt-and-pepper noise without blurring and edge destruction has been reported [2]–[3]. A median filter was also applied for image up-sampling. Recently, a median filter was combined with a neuralnetwork to improve the quality of filtered results. The median filter is a non-linear digital filtering technique [4], often used to remove noise from an image or signal. Such noise reduction is a typical pre-processing step to improve the results of later processing (for example, edge detection on an image). Median filters of both recursive and non-recursive types have been considered in the literature. Recursive median filters were shown to be more efficient than those ofthe non-recursive type. A useful special class of median filters are the separable median filters. These filters are particularly easy to implement, by performing successive operation over the rows and columns of the image.

Bovik, Huang and Munson [5] introduced a generalization of the median filter. They defined an order statistic (OS) filter, in which the input value at a point is replaced by a linear combination of the ordered values in the neighborhood of the point. The class of OS filters includes as special cases the median filter, the linear filter, the α-trimmed mean filter, and the max (min) filter, which uses an extreme value instead of the median. For a constant signal corrupted with in additive white noise, an explicit expression was derived for the optimal OS filter coefficients. Both qualitative and quantitative comparisons suggest that OS filters (designed for a constant signal) can perform better than median and linear filters in some application.

Lee and Kassam [6], introduced another generalization of the median filter, which stems from

robust estimation theory. According to different estimators the *L* filter and *M* filter were proposed, in which the filtering procedure uses a running *L* estimator and an *M* estimator, respectively. Because the *L* estimator uses a linear combination of ordered samples for the estimation of location parameters, the use ofa running *L* estimator for filtering resembles the use of an OS filter. Another variation of median filters is the modified trimmed mean (MTM) filter [7]. This filter selects the sample median from a window centered around a point and then averages only those samples inside the window close to the sample median. MTM filters were shown to provide good overall characteristics [8]. They can preserve edges even better than median filters. The same authors also introduced a double-window modified trimmed mean (DWMTM) filter.

In this filter a small and a large window are used to produce each output point. The small window results in the retention of the fine details of the signal and the large window allows adequate additive noise suppression. The DWMTM [9] filter has good performance characteristics. However, it often fails to smooth out thesignal dependent components. All these previous filters can be easily adapted to 3-D filtering by defining a block mask ($n1, n2, n3$), which can play the role of the traditional window of 2-D filters.

Although median filters preserve edges in digital images, they are also known to remove fine image detail such as lines. For example, 3 × 3 median filters removelines 1 pixel wide, and 5 × 5 median filters remove lines 2 pixels wide. In many applications such as remote sensing and X-ray imaging, this is exceedingly important and efforts have been made to develop filters that overcome the problem. In 1987, Nieminen et al. reported a new class of "detail preserving" filters. These employ linear sub filters whose outputs are combined by median operations. There are a great variety of such filters, employing different sub filter shapes and having the possibility of several layers of median operations [10]. Hence we cannot describe them fully here in the space available. Although these filters are aimed particularly at retention of line detail and are readily understood in this context, they turn out to have some corner preserving properties and to be resistant to the edge shifts that arise when there is a nonzero curvature.

LITERATURE SURVEY

In [1] R.-D. Chen, P.-Y. Chen, and C.-H. Yeh, proposed, it is a word-level filter, storing the samples in the window in descending order according to their values. When a sample enters the window, the oldest sample is removed, and the new sample is inserted in an appropriate position to preserve the sorting of samples. All of them are designed so as to exhibit high scalability and to be easily pipelined for higher working frequencies. Here main drawback was system complexity increases, high-speed computation may cause a considerable increase in power dissipation

In [2] D. Prokin and M. Prokin, proposed the ever-increasing demand for high image quality requires fast and efficient methods for noise reduction. A further novel architecture to calculate the median for a moving set of N integers is the best-known order-statistics filter is the median filter. A method ispresented to calculate the median on a set of N W-bit integers in W/B time steps. Blocks containing B-bit slices are used to find B-bits of the median; using a novel quantum-like representation allowing the median to be computed in an accelerated manner compared to the best-known method (W time steps). The general method allows a variety of designs to be synthesised systematically. Here the main drawback is decrease in time in the operating frequency

In [3] D. S. Richards, proposed Median filters have been proposed for the analysis of speech data and in image processing to enhance the data by smoothing the signal and removing noise. Some new designs are given. The 1-D and 2-D cases are discussed, and recursive median filters are also analyzed. Median filters have been proposed for the analysis of speech data and in image processing to enhance the data by smoothing the signal and removing noise. Each design is analyzed in terms of area, time delay, and concurrency. Some new designs are given. The 1-D and 2-D cases are discussed, and recursive median

filters are also analyzed. It is a word-level filter, storing the samples in the window in descending order according to their values. When a sample enters the window, the oldest sample is removed, and the new sample is inserted in an appropriate position to preserve the sorting of samples. Here the main drawback is power consumption is more.

In [4] S. Marshall, proposed Several DSP algorithms need to remove high-frequency or impulsive noise while preserving edges, Efficient architectural implementation for real-time applications involves a careful VLSI design, which takes into account modularity, regularity, adaptability, scalability, throughput, circuit complexity and fault tolerance. Several DSP algorithms need to remove high-frequency or impulsive noise while preserving edges, e.g., in speech and image processing applications: median filtering has been proved to be more effective for achieving this goal than other filtering techniques. Efficient architectural implementation for real-time applications involvesa careful VLSI design, which takes into account modularity, regularity, adaptability, scalability, throughput, circuit complexity and fault tolerance. Here the main drawback is stability is more.

In [5] E. Nikahd, P. Behnam and R. Sameni, proposed in these numerous registers to store copies of samples in the filter, thereby forming a queue structure to enable the identification of the oldest sample after comparing all the data in the filter. This feature enables the construction of filters with relatively large window lengths with negligible reduction in the maximum clock frequency. Here the main drawback is maximum operating frequency was limited by the filter size.

PROPOSED SYSTEM

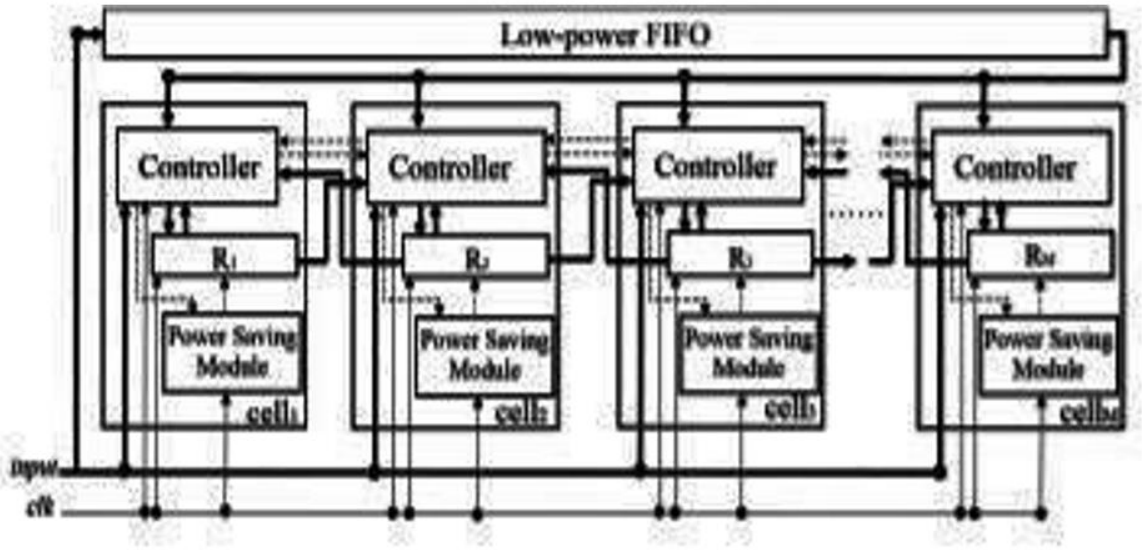
This paper proposes a low-power, high-speed architecture for a 1D median filter. Two techniques are implemented to reduce power consumption. A power-saving module controls the clock signal attached to the components in filter, which gatesthe registers when their values do not need to be changed, particularly for a filter with a large window size. Furthermore, a low-power first-in first-out (FIFO) architecture is designed according to a multi-clock strategy, which not only reduces the power consumption but also achieves high-speed computation. With some extra control units, a more energy- efficient architecture is derived. Both the extensible and real-time median filter networks are regular arrays of bit- wise compare- and- swap and delay units. Also, their internal communication schemes are simple and regular. This makes the VLSI implementations easy and straightforward. The extensible median filter chip can be packaged in a 28-pin package with 2x1 multiplexed I/O's whereas the real-time median filter chip can be packaged in a 40-pin package without multiplexed I/O's. Furthermore, the testing of the VLSI chips may be easily accomplished by the functional test technique, since the operations of the cells can be selectively probed by issuing proper test vectors An overview of the proposed low-power median filter is depicted in Fig. 1. Each cell consists of a controller, power saving module, and data register for storing an input sample. Ri in cell i represents the data register that stores the input sample. The data are arranged in the ascending order after the filter receives a new input and deletes the oldest sample in each cycle. The output of filter is the data in the middle-ranked cell. In a filter with an even length, the output is the average value of data in the (M/2)th and (M/2+1)th cells, which requires an additional adder for summation. The average can be computed through 1bit left shifting. To enhance computation performance, a FIFO architecture is implemented to store copies of all samples in the filter. The output of queue structure FIFO enables the identification of the oldest sample in each cycle. To minimize the power consumption, signal transitions are reduced to their lowest possible level through an energy-efficient algorithm at register-transfer level.In addition, each cell contains a power saving module for controlling the clocksignals attached to the cell. Detailed descriptions of filter are presented in the following subsections. A. Circuit Behavior with the Power- Saving Module A median result is generated after receiving new input from streaming data during each machine cycle. The sample in each cell is updated about the cell

to the right, cell to the left, new input, or original value. These four cases are determined by a set of signals {Ti-1, Ti, Ti+1, Zi-1, Zi}. The detailed architecture of a cell is illustrated in Fig. 2. The signal Ti is the comparison result of the input sample and data in cell i, which is set to 0 when the input is smaller. Ti-1 and Ti+1 are received from the left and right cells, respectively. Ti-1 for the leftmost cell is 0, and Ti+1 for rightmost cell is 1.
The Z signal is the comparison result of current data and the FIFO output, which is asserted to high when the sample in the cell is larger than or equal to the oldest value. Because the filter data are sorted in the ascending order, the Z signals are continuous series of1's followed by series of 0's. The cell containing the data equal to the oldest value canbe determined by using an XOR operator for Zi-1 and Zi, where Zi-1 for the leftmost cell is 0. In the case of more than one cell containing data equal to the oldest value, only one cell is marked, thus denoting that the marked cell may not be the one that exists for the longest period. The four cases of cell updates are listed in Table I. For cells updated by their original values (i.e., no change in data), the power-saving module gates the clock signals attached to the cells to minimize power dissipation.
The frequency at which the power-saving module gates the clock directly depends on the probability of successive blocks containing identical data. This condition can be realized as mutually exclusive events of every two neighboring cells in filters containing different data. The probability is derived using, where w is the bit-width of the samples and M indicates filter size. The sample distribution is considered to be uniform. Each cell has 2w possible values, and the whole possibility of combinations is 2wM. (2w−1) indicates a different value from theprevious one.
Design of low power median filter

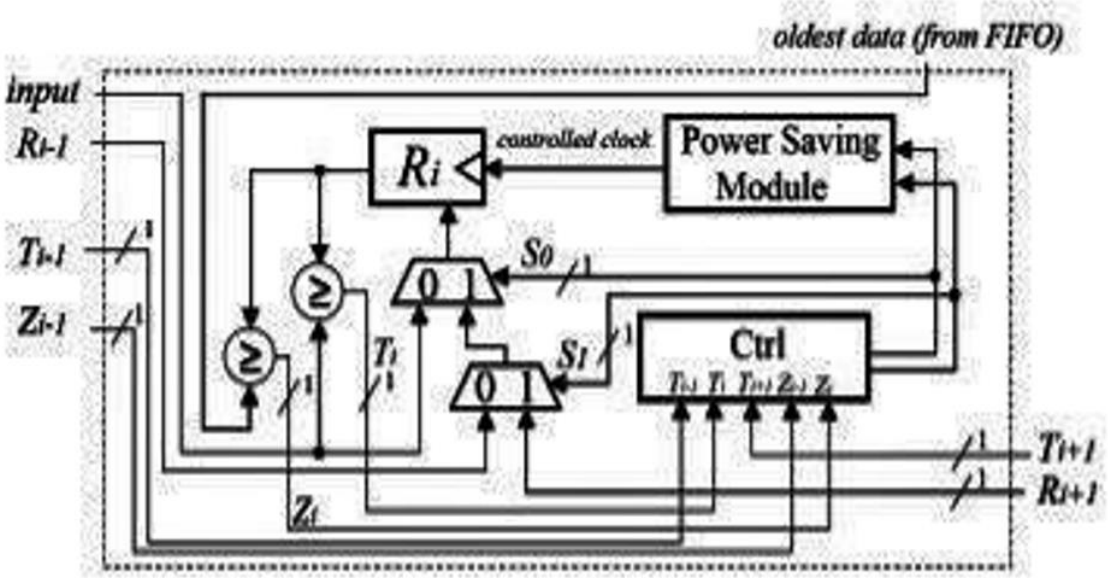
An overview of the proposed low-power median filter is depicted in Fig. 1. Each cell consists of a controller, power saving module, and data register for storing an input sample. Ri in cell i represents the data register that stores the input sample. The data are arranged in the ascending order after the filter receives a new input and deletes the oldest sample in each cycle. The output of filter is the data in the middle-ranked cell. In a filter with an even length, the output is the average value of data in the (M/2)th and (M/2+1)th cells, which requires an additional adder for summation. The average can be computed through 1bit left shifting. To enhance computation performance, a FIFO architecture is implemented to store copies of all samples in the filter.

The output of queue structure FIFO enables the identification of the oldest sample in each cycle. To minimize the power consumption, signal transitions are reduced to their lowest possible level through an energy-efficient algorithm at register-transfer level.In addition, each cell contains a power saving module for controlling the clock signals attached to the cell. Detailed descriptions of filter are presented in the following subsections. A. Circuit Behavior with the Power- Saving Module A median result is generated after receiving new input from streaming data during each machine cycle. The sample in each cell is updated about the cell to the right, cell to the left, new input, or original value. These four cases are determined by a set of signals {Ti-1, Ti, Ti+1, Zi-1, Zi}. The detailed architecture of a cell is illustrated in Fig. 2. The signal Ti is the comparison result of the input sample and data in cell i, which is set to 0 when the input is smaller. Ti-1 and Ti+1 is received from the left and right cells, respectively.

Vol 13, Issue 01, Jan /2022
ISSN NO:0377-9254



Overview of proposed low-power median filter.



. Architecture of the cell in proposed low-power median filter.

Ti-1 for the leftmost cell is 0, and Ti+1 for rightmost cell is 1. The Z signal is the comparison result of current data and the FIFO output, which is asserted to high when the sample in the cell is larger than or equal to the oldest value. Because the filter data are sorted in the ascending order, the Z signals are continuous series of 1's followed by series of 0's. The cell containingthe data equal to the oldest value can be determined by using an XOR operator for Zi-1 and Zi, where Zi-1 for the leftmost cell is 0. In the case of more than one cell containing data equal to the oldest value, only one cell is marked, thus denoting that the marked cell may not be the one that exists for the longest period. The four cases of cell updates are listed in Table I. For

cells updated by their original values (i.e., no change in data), the power-saving module gates the clock signals attached to the cells tominimize power dissipation.
The frequency at which the power-saving module gates the clock directly depends on the probability of successive blocks containing identical data. This condition can be realized as mutually exclusive events of every two neighboring cells in filters containing different data. The probability is derived using, where w is the bit-width of the samples and M indicates filter size. The sample distribution is considered to be uniform. Each cell has 2w possible values, and the whole possibility of combinations is 2wM. (2w−1) indicates a different value from theprevious one.

Table: Truth table for filter behaviour

Case	**0**	**1**	**2**	**3**
Signals	00000	00111	11100	00011
	00100	01100	11101	00001
{Ti-1, Ti, Ti+1, Zi-1, Zi}	11111	00100		
	01111	01101		
Updated by	Origin *	Input	Left	Right

*The clock signal is gated to registers at case 0

RESULTS AND DISCUSSION

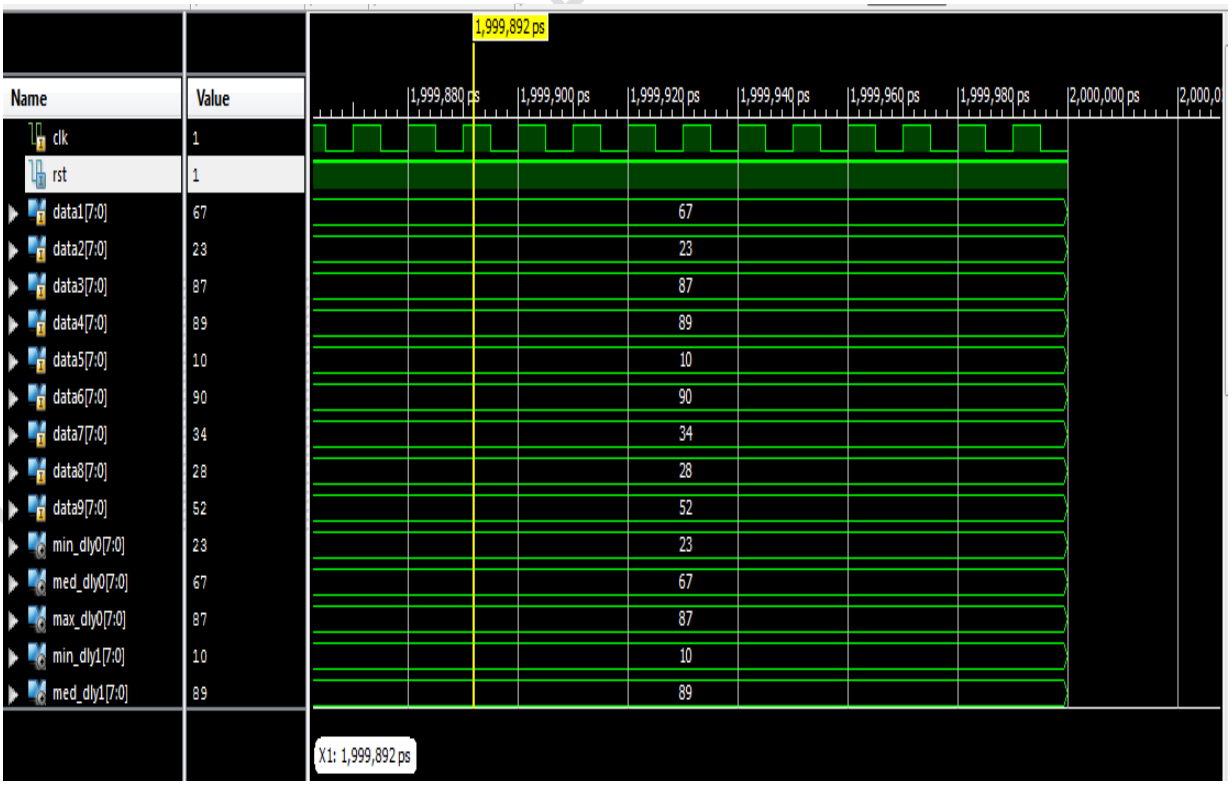


Figure 3: Median filter values when it is in rest position

Here in the above waveform clk is the input to the circuit and it continuously generating '0' and '1'. Data input are data1, data2, data3…… data9 and these eachdata will take 8-bit. Min_dly, med_dly and max_dly are outputs.

Journal of Engineering Sciences
Vol 13, Issue 01, Jan /2022
ISSN NO:0377-9254

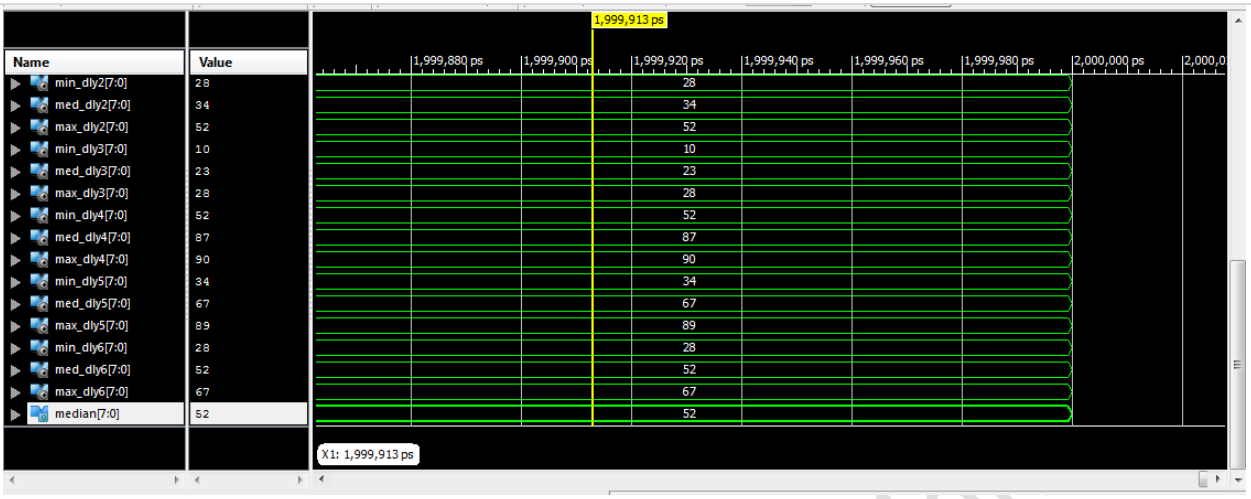


Figure 4: Here median is the output and it show the median data from allgiven input.
INPUT VALUES:
0,7,7,51,51,51,51,51,50,50,50,6,5,5,6,50,50,67,67,67,67,67,68,68,68,83,68,67,67,67,
67,50
,67,67,67,68,68,67,67,68,101,101,84,67,50,50,35,7,7,7,7,7,51,51,51,51,51,50,50,50,
6,55,6,
50,50,67,67,67,67,67,68,68,68,83,68,67,67,67,67,50,67,67,67,68,68,67,67,68,101,10
1,84,6
7,50,50,50,51,51,51,51,51,51,51,51,50,50,50,6,5,5,6,50,50,67,67,67,67,67

OUTPUT VALUE: 52

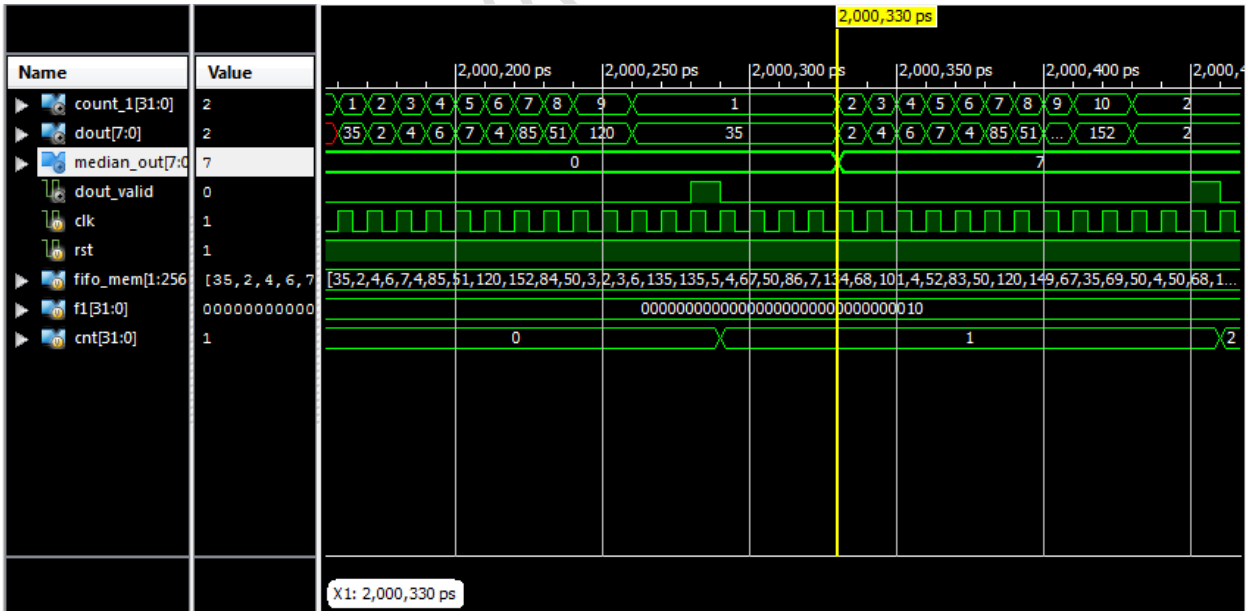


Figure 5 OUTPUT IN THE FORM OF WAVE FORM

Here count_1 is the register and it is a 32-bit. doubt is the input to the median architecture and it will give 8-bit data. Median is the output. It will give output as 8-bit data. dout_valid is a one-bit output. clk and rst are the input of the median architecture. Fif0_mem is the memory which stores 256-bit data. f1 and cnt are registers which stores 32-bit data.

Journal of Engineering Sciences
Vol 13, Issue 01, Jan /2022
ISSN NO:0377-9254

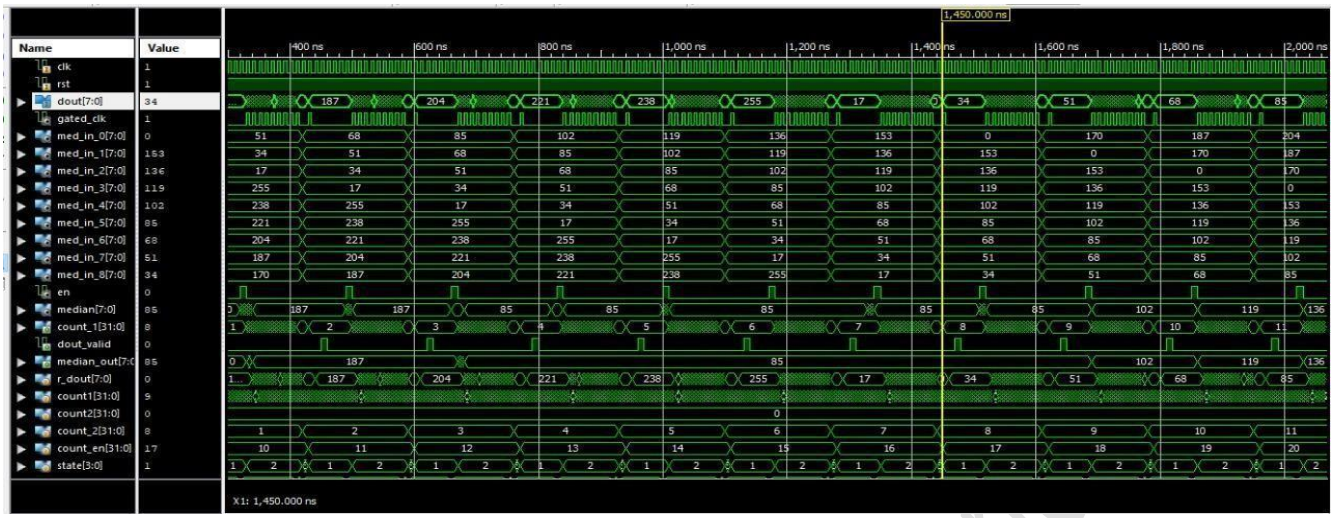


Figure 6: OUTPUT OF MEDIAN FILTER USING CLOCK GATING TECHNIQUE

INPUT VALUES:

0,7,7,51,51,51,51,51,50,50,50,6,5,5,6,50,50,67,67,67,67,67,68,68,68,83,68,67,67,67,
67,50
,67,67,67,68,68,67,67,68,101,101,84,67,50,50,35,7,7,7,7,7,51,51,51,51,5150,50,50,6
,5,5,6
,50,50,67,67,67,67,67,68,68,68,83,68,67,67,67,67,50,67,67,67,68,68,67,67,68,101,1
01,84,
67,50,50,50,51,51,51,51,51,51,51,51,50,50,50,6,5,5,6,50,50,67,67,67,67,67

OUTPUT VALUE: 52

Here in the above wave form clock and rst are the inputs and clock gating is also an input is connected to the memory. Median is the input in the above waveform. End is the enable signal which is one bit. median is the output which gives median number from given input. Count and state are register which are internally declared in the code.

CONCLUSION

This paper proposes a VLSI design and implementation for an energy-efficient median filter. Power dissipation is reduced considerably through a multi-clock strategy and power saving module. The proposed method reduces switching activities and renders the data in the registers immobile, thereby minimizing the total dynamic power. The expected improvement value of switching activity was derived through statistical and probabilistic analyses when assuming uniform distributed data, and the experimental results obtained using 90-nm technology indicated the reliability of the predictions. The average reduction in power dissipation is 25% with an overhead of 1.2 timesmore area cost. The proposed architecture provides an alternative solution for low-power applications, and is considerably more feasible than existing methods for application in wearable and implantable devices In this work an energy efficient median filter architecture has been developed. This architecture provides first by determining the bottle neck on energy efficiency, which has been deduced to memory power. Reduces the memory power consumption using a memory activation scheduling technique. The upper bound for any for any median filter has been estimated using the target platform, which can be used to measure energy efficiency of the technique We achieved up to 53% of the peak energy efficiency and a minimum of 63 framesper second (fps) for the worst case of a 1024x1024 image, which is still above the required 25+ fps.

Vol 13, Issue 01, Jan /2022
ISSN NO:0377-9254

REFERENCES

R.-D. Chen, P.-Y. Chen, and C.-H. Yeh, "Design of an area-efficient one-dimensional median filter," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 60, no. 10, pp. 662–666, Oct. 2013.

D. Prokin and M. Prokin, "Low Hardware Complexity Pipelined Rank Filter," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 57, no. 6, pp. 446–450, June 2010.

D. S. Richards, "VLSI median filters," IEEE Trans. Acoust., Speech, Signal Process., vol.38, no. 1, pp. 145–153, Jan. 1990.

S. Marshall, Logic-Based Nonlinear Image Processing. Bellingham, WA: SPIE Press, 2007, pp. 57–71. [3] G. Mikolajczak and J. Peksinski, "Estimation of the variance of noise in digital images using a median filter," 2016 39th International Conference on Telecommunications and Signal Processing (TSP), Vienna, 2016, pp. 489–492.

E. Nikahd, P. Behnam and R. Sameni, "High-Speed Hardware Implementation of Fixed and Runtime Variable Window Length 1-D Median
Filters," in IEEE Transactions on Circuitsand Systems II: Express Briefs, vol.63, no. 5, pp. 478–482, May 2016.

Shih-Hsiang Lin, Pei-Yin Chen" Hardware design of an Energy-Efficient High Throughput Median Filter" 10.1109/TCSII.2018.2790425, IEEE Transactions on Circuit and Systems

J. Cadenas, G. M. Megson, R. S. Sherratt, and P. Huerta, "Fast median calculation method," Electron. Lett., vol. 48, no. 10, pp. 558–560, May 2012
V. G. Moshnyaga and K. Hashimoto, "An efficient implementation of 1D median filter," in Proc. IEEE Int. MWSCAS, 2009, pp. 451–454.

T. S. Le, N. T. Do and K. Hamamoto, "Speed up temporal median filter and its application in background estimation,"2016 IEEE International Conference on Computing & Communication Technologies, Research, Innovation, and Vision for the Future, Hanoi, 2016, pp. 175–180.

E. Kayalvizhi and N. Sasirekha, "A modified low power architecture for Gabor filter," 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, 2016, pp. 597–600.