Hardware Implementation of Higher Order Median Filter with Effective Edge Preservation

Vanathi M.[1] Dr. A. Kaleel Rahuman[2]
[1]PG Scholar [2]Associate Professor
[1,2]Department of Electronics & Communication Engineering
[1,2]PSNA College of Engineering & Technology, Tamil Nadu, India

Abstract— Design of median filter capable of filtering 36 pixels which has the efficiency of a conventional filter of size 9. This is achieved by dividing the sliding window matrix of a 6X6 matrix into four 3X3 matrix. Main idea is to synchronize all four 3X3 matrix for Median operation so that it can reproduce conventional 6X6 matrix sliding window. Four different mean values are replaced at a time making the processing speed comparatively quicker than conventional 6X6 Median sorting. Filter size is fixed and the Median operation is done through adaptive median sorting algorithm to minimize the processing time. Data driven clock gating techniques are used in the system to reduce switching transition.

Key words: Median Filter, Data Driven Clock Gating, Adaptive Median, Mean Value

I. INTRODUCTION

Median filtering is a type of smoothening process. In image processing an image is segmented into intrinsic blocks, every block is provide with a value depending on the indensity of light energy, these blocks are called pixels. Every pixel has as a value called gray scale value. Depending on the indensity of light energy these gray scale values are assigned [2][3]. Higher the gray scale value higher is the light energy. Main purpose of Median filter is to remove impulse noise, salt and pepper noise and to produce a clear image. Existing architecture goes by word-level architecture, this king of architecture process input pixels sequentially processed and the processed samples are processed in parallel. Very large scale integrated circuits undergoes power crisis main idea of delivering low power to the circuits are breached due to dynamic power consumption. VLSI circuits are functional with clock pulses. These clocking pulses takes most of the power delivered to the circuit. Switching activity in the circuit is initiated by clock pulses so as to register values and signal transition. These switching activity sometimes go redundant making dynamic power dissipation. Usage of FIFOs stores the data as in not deliver the output once the clock pulse is provided. Basically a FIFO is a memory element that can read and write data with respect to clock pulse and enable signal. Values stored in FIFOs are accessed when there is need for that particular data. Existing architecture uses Token Ring architecture [2] with FIFO. Moreover thses FIFOs are designed to generate low power irrespective of type of operation. With 1-D median filtering technique, in this existing architecture size of the filters are adjustable. Through FIFO queuing of input values aren't necessary everything will be stored irrespective of delay in operation.

Median filtering involves two steps one to sort the input pixel and another take the middle value or the median value of the sequence. Exisiting system with the use of FIFO present value and the previous values are compared and stored in a register. When the new value arrives these values stored in FIFO are changed or removed. All these registers are clock gated so as not to take the same pixel again and again. And this stops switching losses, to make this possible power modules which are capable of gating clock pulses to the registers are used. In our proposed method filter size is fixed and our main objective of the project is to provide better edge preservation though filter order increases. With the usage of Adaptive median filtering technique number of registers to store value is less compared to Existing model [8][9] for a filter size of 9. Since it is a sequential sorting this method follows a set of nonlinear comparison pattern hence design of this type of sorting architecture is costlier but efficient. Clock gating is done at the output of Median filter to reduce the switching frequency caused when storing the same mean value.

II. MEDIAN FILTER DESIGN

A. Median Filtering Operation

1	5	189	28	08	11	97	121	32
25	19	33	69	37	33	33	69	37
128	7	78	67	39	26	78	67	39
189	139	91	25	35	67	91	25	35
33	69	37	33	33	69	37	33	33
78	67	39	26	78	67	39	26	78
91	25	35	67	91	25	35	67	91
67	89	48	56	67	89	48	56	67

Fig. 1: Segmented Image

Median sorting is done depending on the location of sliding window. This window is a spatial window of dimension 2D. During window sliding this window covers a particular range of elements depending on the size of filter. After median filtering with the elements it had covered mean value is obtained, this mean value is replaces existing pixels in the matrix. Once the values are replaced window is shifted to another location. This process is done until all the impulsive noises are replaced with mean value. As said before Median filters tend to lose their exclusive properties when the size of filter increases causing to lose edge preservation. In sliding window filtering, the 2D window is defined as an M-by-M square kernel (for odd M) whose center is positioned in the currently handled pixel. M being odd ensures a well-defined center to the window. The window estimate M can be increased in request to expel higher percentages of shot noise, be that as it may, this likewise causes more picture blurring. Therefore, an ordinary 3x3 pixels square window has been chosen in this paper.Fig.1 represents the depicted median substitution process with a 3x3 pixels square window. It ought to be noted that the analyzed "dead pixel" is expelled from the original picture window.

2	2	2	2	2	2	97	121	32
2	2	2	2	2	2	33	69	37
2	2	2	2	2	2	78	67	39
2	2	2	2	2	2	91	25	35
2	2	2	2	2	2	37	33	33
2	2	2	2	2	2	39	26	78
91	25	35	67	91	25	35	67	91
67	89	48	56	67	89	48	56	67

Fig. 2: After Median Sorting

Typical median operation is done by replacing all 36 pixels but doing so will reduce the sharpness in image. Hence this method is replaced by designing an architecture that could remove the constraint making a four 3X3 matrix analysis that will preserve the image

B. Overall Architecture for 6X6 Median Filter

Over all architecture discuss about how 36 pixels can be distributed for Median operation. There are mainly three blocks to implement the objective 1) pixel division 2) feeding pixels to FIFOs for Median sorting 3) obtaining mean value. shown in Fig.2.

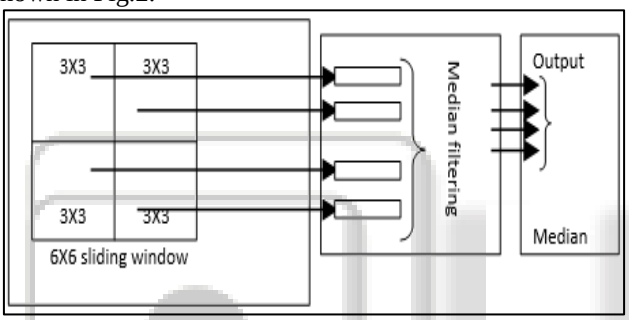

Fig. 3: Overall Architecture of Proposed System

All the blocks are activated through clock pulses. Say it is activated through rising edge triggered clock pulse. For every positive clock pulse each blocks are activated so that coordination of all four 3X3 matrix is maintained. Clock gating is done as a separate part.

Global clock controls all the inputs pulses to all the block in architecture. System is designed in such a way that initially all the pixel values are fed into SRAM and then with the change in clock pulse or through edge triggering window sliding is achieved. As window slides location of pixels in SRAM is assigned as input for Median sorting. Clock gating is also controlled by global clock pulse to reduce switching transition.

C. Sorting Operation

In request to process each picture pixel, the 3x3 square window ought to be traveled through the picture. In any case, in FPGAs, this can likewise be accomplished using a nine-pixel stream that sequentially goes through the median channel. Each gathering of nine pixels can be arranged using a structure made out of two-input exchange nodes as the one delineated in Fig-3.

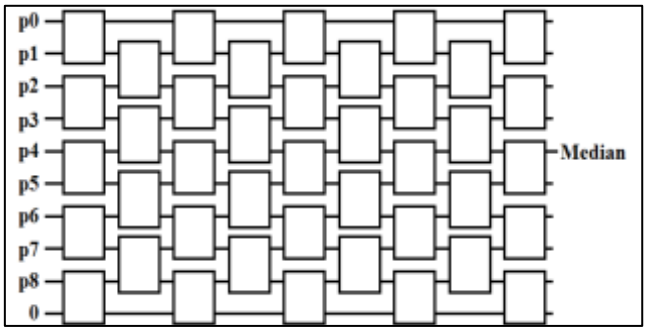

Fig. 4: Conventional Median Sorting Method

The exchange node in Fig.4 plays out a two-input sorting using an 8-bit comparator and two 2:1 multiplexers. The two inputs are internally looked at and the higher (H) and lower (L) values are obtained. The exemplary sorting network structure shown in Fig-4 employment 41 essential nodes. In this figure, each container is an identical exchange node as the one delineated in Fig.5. This exemplary implementation has proven to be a long way from ideal since enhanced FPGA implementations have been produced throughout the years [10][9].

Using the Batcher bitonic sorter [6], a 9-input sorting network can be made [1]. This sorting network, made of 28 fundamental nodes, significantly lessens the amount of FPGA assets utilized. Number of exchange nodes. In this network, the asset utilization minimization is accomplished in light of the fact that it plays out a nine pixel halfway sorting, which means that higher and bring down yield pixel esteems are unsorted. This isn't an issue since only the median filtering is needed to isolate the center pixel. Fig.3 represents this ideal plan.

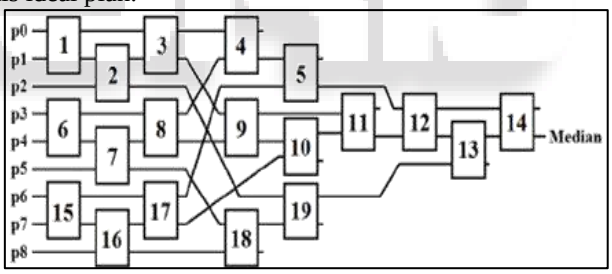

Fig. 5: Adaptive Median Sorting

III. CLOCK GATING METHOD

The clock signal driving a FF is latched (gated) when the FFs state isn't liable to change in the next clock cycle [5]. Data driven gating is causing territory and power overheads that must be considered. In an endeavor to lessen the overhead, it is proposed to assemble a few FFs to be driven by a similar clock signal, Generated by orienting the enabling signals of the individual FFs. this may be that as it may, bring down the disabling effectiveness. It is in this way beneficial to assemble FFs whose switching exercises are exceedingly related and determine a joint enabling signal. In a recent paper, a model for data-driven gating is produced dependent on the toggling action of the constituent FFs [9]. The ideal fan-out of a clock gaiter yielding maximal power savings is inferred In view of the normal toggling measurements of the individual FFs,

Process technology and cell library in utilize. Typical operation the state transitions of FFs in computerized

frameworks depend on the data they process. Assessing the effectiveness of data-driven clock gating requires, in this way, extensive simulations and factual analysis of the FFs' action. Another way is to group of FFs for clock switching dynamic power reduction, called multibit FF (MBFF), has recently been proposed

In [4] and [5]. MBFF endeavors to physically blend FFs into a single cell with the end goal that the inverters driving the clock beat into its lord and slave hooks are shared among all FFs in a gathering. MBFF grouping is mainly driven by the physical position closeness of individual FFs, while grouping for data driven. Clock gating should combine toggling similitude with physical position considerations. Therefore FFs must be placed in a gathering to amplify the power reduction.

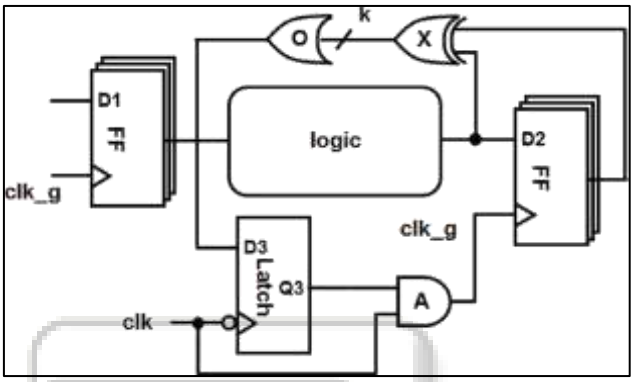

Fig. 6: Data Driven Clock Gating

The data-driven gating proposed in [9] is shown in Fig.6. A FF finds out that its clock can be limited in the Next cycle by XORing its yield with the present data input that will show up at its yield in the next cycle. The yields of d XOR gates are ORed to generate a joint gating signal for d FFs, or, in other words to maintain a strategic distance from glitches. The combination of a lock with AND gate is commonly utilized by business.

Devices and is called integrated clock gate (ICG) [6]. Such data driven gating is utilized for an advanced channel in an ultralow-energy design [10]. A single ICG is amortized over d FFs. Due to this unique ability data driven clock gating techniques are for application oriented system design.

IV. CONCLUSION

Main idea of the proposed system is achieved through adaptive median filtering and data driven clock gating. Speed of operation for 9 input pixel is replicated for a 36 input pixel analysis [1][2]. Hence forth the speed of operation is increased irrespective of order of filter. Only drawback is that filter size is fixed, any new methods to overcome this draw back can be as an extension for this paper. Area, time and switching transition are conserved with proper edge preservation in image.

REFERENCES

[1] A. Farrahi, C. Chen, A. Srivastava, G. Tellez, andM. Sarrafzadeh, "Activity- driven clock design," IEEE Trans. Comput. Aided Des. Integr. Circuits Syst., vol. 20, no. 6, pp. 705–714, Jun. 2001.
[2] C. Chunhong, K. Changjun, and S. Majid, "Activity-sensitive clock tree construction for low power," in Proc. ISLPED, 2002, pp. 279–282.
[3] L. Benini, A. Bogliolo, and G. De Micheli, "A survey on design techniques for system-level dynamic power management," IEEE Trans. VLSI Syst., vol. 8, no. 3, pp. 299–316, Jun. 2000.
[4] M. S. Hosny and W. Yuejian, "Low power clocking strategies in deep submicron technologies," in Proc. IEEE Int. Conf. Integr. Circuit Design Technol., ICICDT 2008, pp. 143–146.
[5] M. Donno, E. Macii, and L. Mazzoni, "Power-aware clock tree planning," in Proc. ISPD, 2004, pp. 138–147.
[6] Synopsys Design Compiler, Version E-2010.12-SP2.
[7] S. Wimer and I. Koren, "The Optimal fan-out of clock network for power minimization by adaptive gating," IEEE Trans. VLSI Syst., vol. 20, no. 10, pp. 1772–1780, Oct. 2012.
[8] S. Wimer and I. Koren, "Design flow for flip-flop grouping in data-driven clock gating," IEEE Trans. VLSI Syst., to be published
[9] V. G. Oklobdzija, Digital System Clocking – High-Performance and Low-Power Aspects. New York, NY, USA: Wiley, 2003.
[10] W. Shen, Y. Cai, X. Hong, and J. Hu, "Activity and register placement aware gated clock network design," in Proc. ISPD, 2008, pp. 182–189.

