V4b.7

DESIGN AND IMPLEMENTATION OF A GENERAL PURPOSE VLSI MEDIAN FILTER UNIT AND ITS APPLICATIONS

Mustafa Karaman, Levent Onural, and Abdullah Atalar

Electrical and Electronics Eng. Dept., Bilkent University
POB. 8, 06572 Maltepe, Ankara, Turkey

ABSTRACT

In order to meet the changing demands of different median filtering applications, a VLSI median filter unit is designed and implemented in 3-μm M^{2}CMOS by employing full-custom VLSI design techniques. The unit consists of two single-chip median filters, one extensible and one real-time. The architectures of the chips are bit-level pipelined systolic structures based on the odd/even transposition sorting. The extensible chip is designed for the applications requiring variable window sizes and variable word-lengths whereas the other one is for real-time applications. Various median filtering techniques are easily realized by using the designed chips together with a reasonable external hardware.

1. INTRODUCTION

The median filtering is a nonlinear smoothing technique that has been frequently used in many signal and image processing applications to filter out the impulsive noises while preserving the edge-information [1]. In the standard median filtering applications, a window of size w, w is odd, moves on the sampled values of the signal or image, and then the median of the samples within the window is computed and written as the output element at the location of the center of the window [2]. In terms of impulsive noise suppression, edge preservation, and ease of design, the performance of median filters are better than the other smoothing filters such as linear filters and generalized mean filters [3].

In order to increase the performance of the median filters for particular applications, various median filtering techniques have been developed. For impulsive noise suppression, the standard median filtering technique is a good choice. However, for suppression of nonimpulsive noises other techniques such as adaptive-length [4], separable [5] recursive [6], and weighted [7] median filtering techniques may be more convenient. For edge detection, generalized [8], hybrid [9], and selective [10] median filtering techniques are frequently used. In addition, the weighted median filtering can be also used for edge detection by choosing the weight coefficients properly.

Since the computation of the median of a group of elements is the fundamental operation in all of the techniques cited above, their realizations can be accomplished using the standard median filter as the basic component. Thus, there have been much efforts to develop high performance software and hardware standard median filters [11,12,13,14]. However, the window size of the median filter and the word-length of the elements are not the same in different applications. Also, the required speed of the filtering operation varies depending on the application. A median filter which is intended as a component of a general purpose signal or image processor must meet these changing demands. We propose a solution to that problem in the form of two single-chip median filters, one extensible and one real-time, which are implemented in 3-μm M^{2}CMOS by employing full-custom VLSI design techniques. The extensible median filter chip is designed for the applications requiring variable word-lengths and variable window sizes whereas the real-time median filter chip is for the real-time median filtering applications. The architectures of the chips are bit-level pipelined systolic structures based on the odd/even transposition sorting. In the following sections, the architectures, VLSI implementations, and some possible applications of the chips are presented.

2. ARCHITECTURES

2.1 Extensible Median Filter Architecture

The extensible median filter is an odd/even transposition sorting network which is a bit-level pipelined regular structure consisting of 9 *compare-and-swap* stages (Fig.1). Each stage consists of 5 bitwise *compare-and-swap* units. Each of these units serially compares two numbers at its inputs and interchanges them if necessary so that the larger one is at the "top". At the output of the last stage, the data will be sorted such that the largest will be at the top, and the median will be in the middle. At each clock, one bit from each word (total of 9 bits) enter the network and one bit of the median is obtained at the output. The flow is from the most significant bits toward the least significant bits both at the input and at the output. Because of the bitwise serial data flow, this structure allows arbitrary word-length, L.

The bitwise compare-and-swap unit (CSU1) is a finite state machine which has three legal operation states: *equal* ($SE = 01$), *pass* ($SE = 00$), and *swap* ($SE = 10$) (Fig.2). CSU1 is set to the equal state at the end of each data word by a reset signal. Thus the reset signal flows through the stages of the network at a rate of one stage per clock cycle by means of the pipelined delay units. During the computation, the CSU1 stays in the equal state and passes the input data unaltered as long as the two input bits are equal as they flow in. However, it locks itself into the pass state when it first finds that $A_i > B_i$ and passes the inputs unaltered. On the other hand it locks itself into the swap state when it first finds that $A_i < B_i$ and swaps the inputs.

CH2673-2/89/0000-2548 $1.00 © 1989 IEEE

In the extensible median filter structure given in Fig.1, the *upper* and *lower extension* I/O's ($x_{i/o}$'s and $y_{i/o}$'s) are used to extend the filter to larger window sizes. For $w = 9$, the upper and lower extension inputs are connected to logic 1's and logic 0's so that the corresponding compare-and-swap units act as delay units. On the other hand, the design allows the interconnections of many of these chips to form median filters for $w > 9$ (Fig.3).

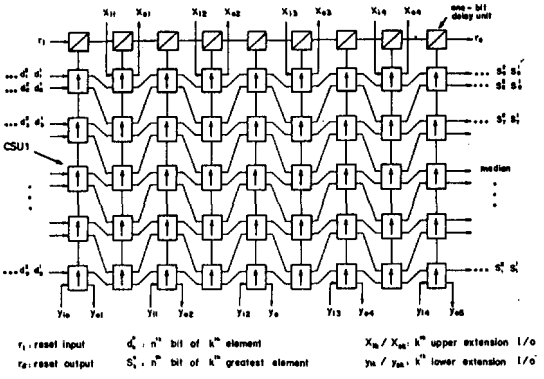

Figure 1: The extensible median filter architecture.

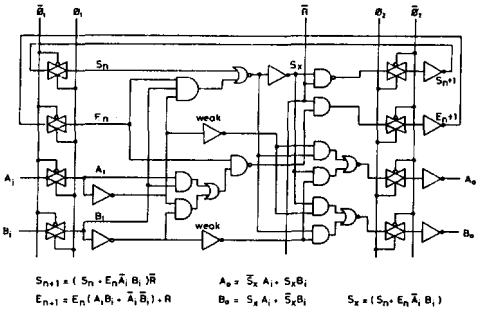

Figure 2: Logic diagram of the compare-and-swap unit-1 (CSU1).

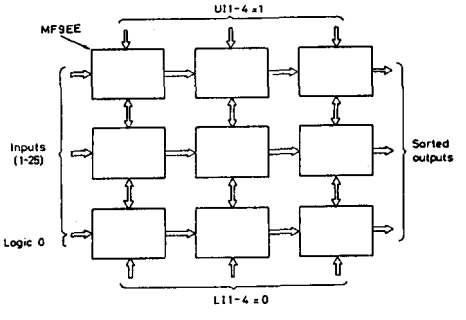

Figure 3: Interconnections of the extensible median filters for $w = 25$ (MF9EE: extensible median filter with window size $w = 9$).

The extensible median filter generates its outputs with a delay of $w + L$ clocks; and after the network is full, it finds one L-bit median per L clocks. Although, the resulting speed may be sufficient for the real-time median filtering of 512×512 frames with $L < 5$, it is not enough for the real-time filtering of 1024×1024 frames with $L > 1$.

2.2 Real-Time Median Filter Architecture

The real-time median filter is designed by interconnecting 8 odd/even transposition sorter blocks in parallel [13] (Fig.4). In this network, the data enter in such a way that the most significant bits go to the first block, the second most significant bits to the second block, and so on. The bitwise compare-and-swap unit used in this network is slightly different than that of the extensible one, because the"swap" or "pass" information flows from upper to lower blocks so that the compare-and-swap unit takes this information, uses, updates and sends it out (Fig.5). For proper timing, the pipelined delay units are included at the input and output of the network.

The real-time median filter has nine 8-bit data inputs and it generates one 8-bit median per clock. At every clock, three new elements enter the chip, corresponding to the new elements of a sliding 3×3 window. Since the clock period is determined by the delay of one compare-and swap unit (CSU2), recent VLSI technology allows the implementation of CSU2 at a speed larger than the real-time operation rate for the 1024×1024 frames with $L = 8$.

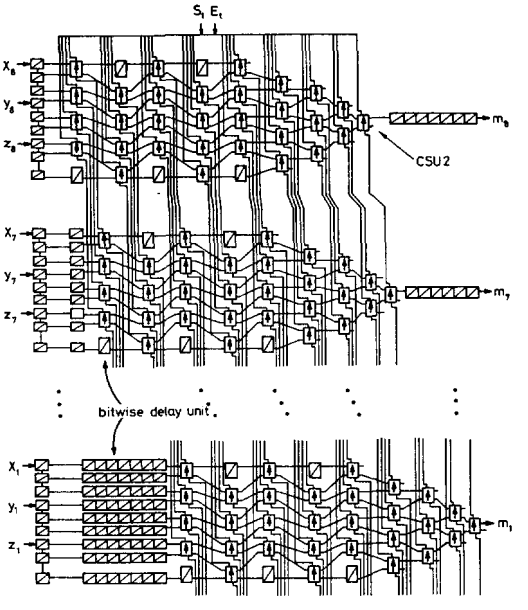

Figure 4: The real-time median filter architecture.

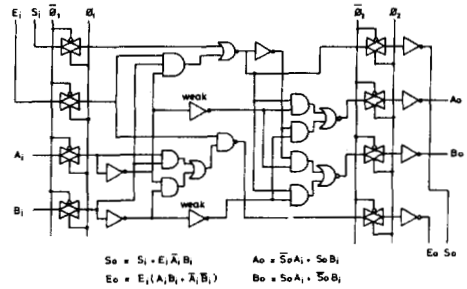

Figure 5: Logic diagram of the compare-and-swap unit-2 (CSU2).

3. CHIPS

Both of the extensible and real-time median filter architectures are regular arrays of the bitwise compare-and-swap units. Also, their internal communication schemes are simple and regular. This makes the VLSI implementations easy and straightforward [15,16]. The architectures are mapped to hardware by using standard CMOS logic style [17] in 3-μm double metal n-well process. For generation of the chip layouts, and their simulations, full-custom VLSI CAD tools [18,19] are used: *magic* for layout editing, *Spice, Rnl*, and *Esim* for simulations. The overall layouts of the chips are shown in Fig.6. The main features of the chips are given in Table.1.

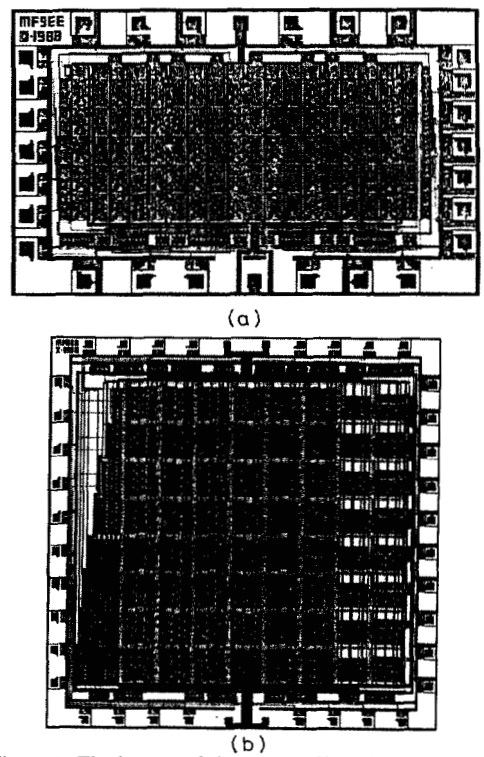

Figure 6: The layouts of the median filter chips: **a)** extensible, **b)** real-time.

Parameter	Extensible	Real-Time
Die size (mm^2)	2.8×4.5	7.2×6.9
Transistor count	5,000	22,000
Transistors/mm^2	400	450
Maximum clock frequency (MHz)	40	50
Maximum throughput (mega medians/sec)	$30/L$	50
Pin number	28	40
Maximum power dissipation (mW)	250	800

Table 1: The main features of the chips.

The testing of the chips are easily accomplished by the *functional* test techniques [20] since the operations of the cells can be selectively probed by using proper test vectors. The test vectors and the expected outputs are generated by using software tools written for these purposes. There are 500 test vectors for the extensible median filter chip, and 12,000 for the other one.

4. APPLICATIONS

The extensible and the real-time median filter chips can be selectively used in a processor environment by means of the chip enable signal that each chip has. Furthermore, one can realize any median filtering technique mentioned in the introduction by using the extensible and/or the real-time median filter chips together with or without a reasonable external hardware:

- For the standard median filtering technique, the exact medians of the elements, in a window size $w = 9$ with arbitrary word length L, can be found by using only one extensible median filter chip. For $w > 9$ with arbitrary L, at most $\lceil w/9 \rceil^2$ ($\lceil . \rceil^2$ indicates the smallest greater integer) chips are required to find the exact medians (Fig. 3). On the other hand, the real-time median filter chip can find the exact running medians of the elements in a window of a fixed size $w = 9$ with fixed word length $L = 8$ at the real-time rate.

- The extensible median filter is a favorable choice to realize the *adaptive-length* median filters [4], since one can change the window size from 3 to indefinitely large ones by using the extensible median filter chip(s) by applying logic 0's or 1's to unused inputs of the chip(s) appropriately.

- For the realizations of the *weighted* median filters [7], the extensible median filter can be used together with a pipelined multiplier which multiplies the input data with the weight coefficients. Since all input data of the chip are entered to the chip directly at each move of the window, one can realize an adaptive weighted median filter by changing the weight coefficients at each position of the window on the frame.

- A pair of the extensible or the real-time median filter chips can be used as a *selective median* filter [10] together with an external control logic consisting of two full-word subtracter and a full-word comparator.

- Either the extensible or the real-time median filter chip can be used as a *line-recursive* median filter [4] by loading the window elements from the frame appropriately.
- The chips can be used for the realizations of the *separable* median filters [5] without any external hardware.

5. CONCLUDING REMARKS

A VLSI median filter unit consisting of two single-chip median filters and its applications are presented. The architectures of the chips are modular and have regular communication schemes which make the VLSI implementations rather easy and straightforward. Both of the architectures are not preferable to be implemented at larger window sizes since the area is proportional to the w^2. We have chosen $w = 9$, because this is the most commonly used window size in two dimensional median filtering applications.

The main contributions of this study are design and implementations of two single-chip versatile components for signal and image processing: an extensible median filter chip for adaptive-word-length and adaptive-window filtering applications, and a real-time median filter chip for real-time filtering of images with sizes up to 1024×1024 pixels. Furthermore, it is concluded that a general purpose median filter unit can be formed by selectively using the chips in a full-scale general purpose digital signal or image processor environment.

ACKNOWLEDGMENT

This research was sponsored by NATO's Scientific Affairs Division in the framework of the Science for Stability Programme.

References

[1] J. W. Tukey, "Nonlinear (nonsuperposable) methods for smoothing data," in *Conf. Rec.*, p. 673, *EASCON* 1974.

[2] N. C. Gallagher, Jr., and G. L. Wise, "A theoretical analysis of the properties of median filters," *IEEE Trans. Acoustic, Speech, and Signal Processing*, vol. ASSP-29, pp. 1136-1141, Dec. 1981.

[3] A. Kundu, S. K. Mitra, and P. P. Vaidyanathan, "Application of two-dimensional generalized mean filtering for removal of impulse noises from images," *IEEE Trans. Acoustic, Speech, and Signal Processing*, vol. ASSP-32, NO. 3, pp. 600-609, Jun. 1984.

[4] H. M. Lin and A. N. Willson, Jr., "Adaptive-length median filters for image processing," Proc. of IEEE ISCAS'88, pp. 2557-2560.

[5] T. A. Nodes and N. C. Gallagher, Jr., "Two-dimensional root structures and convergence properties of the separable median filter," *IEEE Trans. Acoustic, Speech, and Signal Processing*, vol. ASSP-31, pp. 1350-1365, Dec. 1983.

[6] C. G. Boncelet Jr., "Recursive algorithms and VLSI implementations for median filtering," Proc. of IEEE ISCAS'88, pp. 1745-1747.

[7] T. Loupas, W. N. McDicken, and P. L. Allan, " Noise reduction in ultrasonic images by digital filtering, " *The British Journal of Radiology*, vol. 60, pp.389-392, Apr. 1987.

[8] Y. H. Lee and S. A. Kassam, "Generalized median filtering and related nonlinear filtering techniques," *IEEE Trans. Acoustic, Speech, and Signal Processing*, vol. ASSP-33, pp. 672-683, Jun. 1985.

[9] Y. Neuvo, P. Heinonen, and I. Defee, "Linear-median hybrid edge detectors," *IEEE Trans. Circuits and Systems*, vol. CAS-34, pp. 1337-1343, Nov. 1987.

[10] S. J. Ko, Y. H. Lee, and A. T. Fam, "Selective median filters," Proc. of IEEE ISCAS'88, pp. 1495-1498.

[11] D. L. Knuth, *The Art of Computer Programming-Searching and Sorting*, vol. 3. Reading MA: Addison-Wesley, 1973.

[12] E. Ataman, V. K. Aatre, and K. M. Wong, "A fast method for real-time median filtering," *IEEE Trans. Acoustic, Speech, and Signal Processing*, vol. ASSP-28, pp. 415-421, Aug. 1980.

[13] K. Oflazer, "Design and implementation of a single-chip 1-D median filter," *IEEE Trans. Acoustic, Speech and Signal Processing*, vol. ASSP-31, pp. 1164-1168, Oct. 1983.

[14] V. V. B. Rao and K. S. Rao, "A new algorithm for real-time median filtering," *IEEE Trans. Acoustic, Speech, and Signal Processing*, vol. ASSP-34, pp. 1674-1675, Dec. 1986

[15] M. J. Foster and H. T. Kung, "The design of special purpose VLSI chips," *IEEE Computer*, pp. 26-40, Jan. 1980.

[16] H. T. Kung, "Why systolic architectures?," *IEEE Computer*, pp. 37-46, Jan. 1982.

[17] N. Weste and K. Eshraghian,*Principles of CMOS VLSI Design*, Reading MA: Addison-Wesley, 1985.

[18] *Berkeley CAD Tools User's Manual*, EECS Dep., University of California at Berkeley, 1986.

[19] *VLSI Tools Reference Manual*, TR#87-02-01, Release 3.1, NW Lab. Int. Sys., Dep. Computer Sci., University of Washington, Feb.1987.

[20] J. A. Abraham and W. K. Fuchs, "Fault and error models for VLSI," *IEEE Proc.*, vol. 74, pp. 639-654, May 1986.