Design of a Switched-Current Median Filter

C. K. Tse and K. C. Chun

Abstract—This paper discusses the application of switched-current techniques to the implementation of median filters. A new switched-current element is introduced which has the ability to compare and toggle two signals. Using this new element and other basic constituents such as current mirrors and delay cells, a systolic algorithm can be implemented which performs the signal processing function of median filtering. Median filters implemented in this way enjoy all the inherent advantages of the current-mode circuit operation.

I. INTRODUCTION

The growing use of median filters in smoothing and suppression of speckle noise in image processing, edge detection, feature extraction, signal coding, etc., has stimulated interest in the hardware implementation of median filters [1], [2]. More recently, potential applications are identified of median filters in the processing of signals associated with the control of power electronics systems, since in such systems waveforms carrying sharp edges are often corrupted by impulse-like switching noise [3].

Median filtering, unlike other filtering functions, involves simply an extraction of the statistical median from a set of consecutively sampled values. Comparison and sorting are therefore the bread-and-butter operations. Use of DSP, coupled with software programming, has been predominant in the implementation of such operations. In the case of hardware VLSI implementation using the systolic approach [1], median extraction is realized digitally, and as such its performance is greatly affected by flip-flop delays, limited bit-size (resolution), etc.

Current mirroring, although not always recognized as such, is a natural candidate for comparison and sorting. In this paper we present a new approach to realizing the median filter based on some switched-circuit building blocks. Advantages of the proposed current-mode implementation include high operating speed, high resolution, wide dynamic range, low supply-voltage requirement, simple circuitry, and ease of expansion (due to the use of systolic structure). Furthermore, as a result of the simple configuration and the current-mode operation, standard CMOS process can be used in actual implementation, thus making mixed analog and digital signal processing on a single chip possible [4], [5].

II. THE CURRENT-MODE COMPARE-AND-SWAP (CMCS) ELEMENT

In median filtering, signals are required to compare themselves and shuffle their positions according to their relative rankings. We begin here with the basic current-mode compare-and-swap (CMCS) element which operates on two currents, and whose purpose is to ensure that the two currents are always sorted in either ascending or descending order. Referring to Fig. 1, the outputs of the CMCS are

\[ I_{o1} = \min\{I_1, I_2\}, \quad \text{and} \quad I_{o2} = \max\{I_1, I_2\}. \]

A. Operation of the CMCS Element

Current mirrors are used to duplicate the two input currents for use in the COMPARE operation and in the SWAP operation. In Fig. 1, two types of current mirrors are used: MIRRORA produces one copy of the input current, and MIRRORB produces two copies of the input current. Both mirrors provide a buffering function to the next stage. Fig. 2 shows these circuits.

As shown in Fig. 1, COMPARE determines which current is larger, and produces a current signal \( I_d \) which is then buffered by a mirror and converted to a voltage signal by a sensing circuit. The resulting voltage signal from SENSE is used to control the SWAP element. SWAP toggles the two input currents if the SENSE output \( S \) is high. When \( S \) is low, SWAP simply duplicates the two currents to the outputs. In summary, the CMCS element performs a swapping function whenever \( I_1 > I_2 \). Hence, \( I_{o1} \) is always less than \( I_{o2} \).

B. COMPARE—Current Comparator

The current comparator is shown in Fig. 3 whose operation is explained as follows. The mirroring action of \( M_{11} \) and \( M_{11} \) gives

\[ I_1 = I_{11} + I_p. \]

Similarly, mirroring action forces \( I_2 = I_{12} + I_p \). Also, KCL requires that

\[ I_d = I_2 - I_1 = (I_{11} + I_p) - (I_{12} + I_p). \]

From setting \( I_{11} = I_{12} \), we have

\[ I_d = \begin{cases} I_p - I_o & \text{if} \ I_p > I_o \\ I_o & \text{if} \ I_p = I_o \\ I_p & \text{if} \ I_p < I_o \end{cases} \]

(1)

Manuscript received May 17, 1993; revised December 9, 1993. This work was supported in part by Hong Kong Polytechnic Research Committee under Grant UPOCDIA.340.814.A3.420. This paper was recommended by Associate Editor C. Tomazou.

C. K. Tse is with the Department of Electronic Engineering, Hong Kong Polytechnic University, Hung Hom, Hong Kong.

K. C. Chun is with Valery Limited, Fo Tan, Shatin, N.T., Hong Kong. IEEE Log Number 9409257.

1057-7130/95$04.00 © 1995 IEEE
C. SENSE—Current Sensing Amplifier

Fig. 4 shows the current sensing amplifier whose function is to produce a voltage signal according to the sign of the input current. The circuit is essentially an inverter with its input clamped by the diodes to either ground or the supply voltage. Since the gate capacitance of $M_1$ and $M_2$ is small, a minute current flowing in or out the inverter will cause a voltage to be developed at the input. Transistors $M_3$ and $M_4$ form another inverter which shapes the output voltage so that it will have a symmetric rise and fall time. The logic for the sensing amplifier is therefore

$$V_{out} = \begin{cases} 
\text{High} & \text{if } I_{in} > 0 \\
\text{Low} & \text{if } I_{in} < 0 \\
\text{Unchanged} & \text{if } I_{in} = 0.
\end{cases} \quad (2)$$

Note that if $I_{in} = 0$, $V_{out}$ takes whatever value it previously was. Such a situation rarely happens in practice.

D. SWAP—Current Swapping Circuit

Shown in Fig. 5 is the current swapping circuit which employs NMOS differential current switches to achieve fast settling time. Swapping action can be achieved by driving the gates of $M_2$ and $M_4$ to high via the transmission gate $X_3$, and at the same time pulling the gates of $M_1$ and $M_3$ to ground by driving $M_5$ to conduction. Likewise, driving $M_1$ and $M_3$, while pulling off $M_2$ and $M_4$, will cause no swapping action to be performed. The transmission gates are driven by a pair of complementary signals $SS$ and $SSB$ which are derived from $S$ through chains of inverters, as shown in Fig. 5. The inverters, i.e., $X_1 \cdots X_5$, have different propagation delays, so that the rising and falling transitions of $SS$ and $SSB$ can be made to overlap perfectly. Usually, delay equalization is achieved by choosing appropriate device sizes for the inverters.

E. FDELAY—Full-Clock Delay Element

Since median filtering involves sampling the input periodically, a full-clock delay circuit is needed to sample the signal and to synchronize the operations of the CMCS elements. The need for FDELAY will become apparent when we discuss the systolic algorithm in Section III. The simple first-generation switched-current delay cell [6], as shown in Fig. 6, may be used for this purpose. It should be noted that in simulating this delay cell, for reasons of speed and feasibility, a SPICE macro-model is used that performs sampling and delay to the effect of the actual FDELAY.

F. SPICE Simulation of the CMCS Element

Simulation of the CMCS element is performed using the general purpose simulator SPICE. We have chosen appropriate device sizes for the MOS transistors to minimize clock-feedthrough and to achieve delay equalization. Specifically, all PMOS's and NMOS's in MIROR1, MIROR2, and the current comparator have $W/L = 20 \mu m/1.2 \mu m$. All current sources are $2 \mu A$, and voltage swing is from 0 to 3V. For the current sensor, device sizes of $M1$ and $M2$ are chosen to optimize sensitivity whereas device sizes of $M3$ and $M4$ are chosen to balance the rise and fall times. In our simulation, $W/L$ for devices in $M1$, $M2$, $M3$, and $M4$ of the current sensor are $3.0 \mu m/1.2 \mu m$, $1.2 \mu m/1.2 \mu m$, $9.0 \mu m/1.2 \mu m$, and $4.0 \mu m/1.2 \mu m$, respectively. Finally for the current swapping circuit, device sizes are chosen such that inverter X4 has a propagation delay twice that of X1, X2, X3, and X5, thus ensuring equal delays from S to SS and from S to SSB. X6 is made identical to X7, and X8 is made identical to X9. For X6 and X7, device sizes are $2.4 \mu m/2.4 \mu m$ for PMOS's and 6.0 $\mu m/1.2 \mu m$ for NMOS's. For X8 and X9, device sizes are 9.0 $\mu m/1.2 \mu m$ for PMOS's and 6.0 $\mu m/1.2 \mu m$ for NMOS's. Simulation results from SPICE3c1 are shown in Figs. 7 and 8.
III. SYSTOLIC ALGORITHM

In the preceding section, all elements necessary for the construction of the median filter are discussed. In this section we describe how these elements are put together to make the median filter. The algorithm discussed here uses the odd/even transposition sort as the high level algorithm [2], which was originally used in digital sampled data systems. Essentially, it consists of an input delay stage which generates the successive window elements from the incoming sample stream, and a pipelined sort stage which performs the odd/even transposition sort on the elements of the successive windows. Fig. 9 depicts the structure of this median filter algorithm, with window size equal to 5. It is not difficult to verify that after an initial delay of 5 sampling periods which is required to fill the pipeline, the system will produce one median over a sliding window of width 5 every clock period.

IV. THE SWITCHED-CURRENT MEDIAN FILTER

With the elements developed in Section II and the systolic algorithm explained above, the construction of the switched-current median filter will be a straightforward matter. We now demonstrate the feasibility of using switched-current techniques in constructing rank-based filters by presenting the SPICE simulation of the circuit of Fig. 9.

The circuit is simulated with SPICE3c1. Two types of input signals are used: 1) a corrupted triangular ramp signal; and 2) a corrupted square-wave signal. As can be intuitively reasoned, the choice of window size and sampling rate will greatly affect the smoothing property of the median filter. In our examples, the sampling period is 100 ns and window size is 5. Such choice will ensure that the 1.7 MHz ripples and impulsive ringings are removed while the characteristic shapes of the ramp and the square wave are retained. Performance of our proposed median filter, as reported by SPICE3c1, is shown in Figs. 10 and 11 which clearly illustrate the ability of the median filter in rejecting the unwanted noise.

V. CONCLUSION

In this paper a novel switched-current median filter is introduced, in which a new circuit element, called the current-mode compare-and-swap (CMCS) element, is the critical constituent. Apart from its use in median filters, the CMCS element should find applications in many other signal processing functions that require determining the rank of a signal among a sample of signals. For example, it can be used to realize membership function classifiers in a fuzzy control system. It must be emphasized that the purpose here is to demonstrate the basic principle rather than to present a working version of switched-current median filter. For purpose of illustration and ease of simulation, we have employed exclusively first-generation SI circuits and simple mirrors. In fact, refinements are needed in a number of ways before the circuit can be put to practical use. For example, practical circuit styles, such as second-generation SI circuits and precise mirrors, may be used in lieu of the first-generation circuits and simple mirrors [5]. Nevertheless, in the authors' opinion, switched current techniques, coupled with the current-mode circuit concept, will provide an excellent alternative to the realization of many rank-based filter functions.
An Efficient Weighted Least-Squares Design of Linear-Phase Nonrecursive Filters

S. Sunder

Abstract—An accelerated procedure for the design of linear-phase nonrecursive filters using a weighted least-squares technique is described. This procedure is based on formulating the error reflecting the difference between the desired amplitude response and the amplitude response of the practical filter in a quadratic form. The coefficients of the filter are obtained by solving a system of linear equations involving a Toeplitz-plus-Hankel matrix. Such a system of linear equations can be solved by computationally efficient algorithms having only $O(N^2)$ complexity. By choosing the appropriate frequency-dependent weighting function, a filter with either a least-squares or an equiripple error variation can be designed.

I. INTRODUCTION

Nonrecursive filters can be designed using the Fourier-series method, the frequency-sampling method [1], the Chebyshev polynomial approximation [2], [3] or the least-squares methods [4]–[7]. Of these methods, the Chebyshev polynomial approximation based on the Remez exchange algorithm and the least-squares methods have received considerable attention. The method based on the Remez exchange algorithm is essentially a multivariable optimization technique in which the maxima of an error function are minimized to yield filters that are optimal in a minimax sense. The least-squares methods are based on formulating the weighted mean-square error as a quadratic function. The filter coefficients are obtained by solving a system of linear equations or by evaluating the eigenvector corresponding to the smallest eigenvalue of a real, symmetric and positive-definite matrix. The minimax or least-squares optimality depends on the application at hand.

Because of the lack of existence of the alternation theorem (the theorem used in the Remez exchange algorithm) in two and higher dimensions, weighted least-squares (WLS) techniques for the design

REFERENCES


of equiripple filters have gained considerable attention in the recent past [8]–[11]. The primary reason for the choice of WLS techniques is that the best Chebyshev approximation is also the best WLS approximation provided the least-squares frequency-dependent weighting function that yields an equiripple design is suitably chosen [12]. In [8], a double adaptive systems approach has been used to obtain a suitable real-valued cost function of the least-mean squares (LMS) approximation to design an equiripple filter that is optimum in the Chebyshev sense. In [9], Lawson’s algorithm [13] has been modified to design one- and two-dimensional nonrecursive filters. The method proposed in [10] is similar to that presented in [11] and is also based on a modified LAWSON’s algorithm.

In [14], a WLS method based on a quadratic error formulation in conjunction with the weighting function update technique of [11] was used to design digital differentiators and Hilbert transformers. In this method, the filter coefficients have been obtained by solving a system of linear equations, involving a real, symmetric and positive-definite matrix, using the Cholesky decomposition technique. The WLS method in [14] is considerably faster than that in [10] and [11] since the size of the matrix involved in [14] depends only the length of the filter while that of the matrices involved in [10] and [11] depends both on the length of the filter and the number of sample points.

In this correspondence, we discuss an efficient method for the design of linear-phase nonrecursive filters. The objective function formulation of the design problem is the same as that presented in [14]. Here, however, we exploit the fact that the system of linear equations actually involves a Toeplitz-plus-Hankel matrix. Such a system of linear equations can be solved by computationally efficient algorithms involving only $O(N^2)$ complexity [15]–[18]. Consequently, a significant reduction in the computational complexity as compared to that in [14], and in turn, in [10] and [11], is achieved.

By choosing the frequency-dependent weighting function as in [7], filters that are optimal in a least-squares sense can be designed. On the other hand, equiripple filters can be designed by using the iterative techniques proposed in [10]–[11] and [14].

II. ERROR FUNCTION FORMULATION AND MINIMIZATION

Nonrecursive linear-phase filters with $N$ taps having an impulse response $h(n) (n = 0$ to $N - 1)$ can be divided into four types $[1]$ as described below. Type 1 and Type 2 filters include those with symmetrical impulse responses ($h(n) = h(N - 1 - n)$). Type 1 filters have an odd number of taps while Type 2 filters have an even number of taps. The frequency response of these filters can be expressed as $H(e^{j\omega}) = M(\omega)e^{-j(N-1)/2\omega}$ where

$$M(\omega) = \left\{ \begin{array}{ll}
\sum_{n=0}^{(N-1)/2} b(n) \cos n\omega & \text{Type 1} \\
\sum_{n=1}^{N/2} b(n) \cos (n - 1/2)\omega & \text{Type 2}.
\end{array} \right. \quad (1)$$

If $N$ is odd, $b(0) = h((N - 1)/2)$ and $b(n) = 2h((N - 1)/2 - n)$ for $1 \leq n \leq (N - 1)/2$. If $N$ is even, $b(n) = 2h(N/2 - n)$ for $1 \leq n \leq N/2$.

On the other hand, Type 3 and Type 4 filters have an antisymmetrical impulse response ($h(n) = -h(N - 1 - n)$). Type 3 filters have an odd number of taps while Type 4 filters have an even number of taps. Consequently, the frequency response of these filters can be

1057-7130/95$00.00 © 1995 IEEE