Parallel, Asynchronous, Current-Mode Median Filter Based on Resistive Voltage Divider

Tomasz Talaśka
Faculty of Telecommunication, Computer Science and Electrical Engineering
UTP University of Science and Technology
Bydgoszcz, Poland
Aptiv Services Poland S.A.
Kraków, Poland
tomasz.talaska@gmail.coml

Rafał Długosz
Aptiv Services Poland S.A.
Kraków, Poland
Faculty of Telecommunication, Computer Science and Electrical Engineering
UTP University of Science and Technology
Bydgoszcz, Poland
rafal.dlugosz@gmail.com

***Abstract*—The paper presents a novel non-linear median filter operating in the current mode. The filter consists of an analog and a digital part. A first layer of the analog part converts particular input signals, represented by currents, into 1-bit digital signals delayed by time periods inversely proportional to the values of the currents. The second block of the analog part of the system is a multi-branch resistive voltage divider, switched over by described digital signals. A digital block of the filter is used to determine the address of the input signal that in a given calculation cycle has a median value among all signal samples. The circuit was designed in the 180 nm CMOS technology and verified by comprehensive HSpice simulations. For the input currents up to 50 nA it offers data rate of 100 kSamples/s, dissipating power of 100 μW.**

***Index Terms*—Nonlinear filters, median filter, CMOS technology, parallel data processing, asynchronous circuits, current-mode**

I. INTRODUCTION

Median filters, along with the Min and the Max filters [3], [4] create a group of nonlinear filters commonly used in image and signal processing. Filters of this type have low-pass properties. Their operation can therefore be compared to linear low-pass finite impulse response (FIR) filters [1], [2]. The FIR filters calculate a weighted average of a given number of subsequent samples of the input signal. In some situations, e.g. when an input signal is contaminated by a pepper noise, the FIR filters cause a blurring effect of the noise samples, instead of removing them. In such situations median filters, which are able to remove the noise, may be a better choice. However, in this case a relationship between the median filter length and the width of the inclusions of the noise is important.

The median filtering is based on sorting a given number of subsequent samples of the input signal, and then selecting the central value from the sorted data set. In this way, even if some of the samples deviate significantly from an average / "typical" value in a given set, they are effectively filtered out without the described blurring effect.

Filters of this type may be implemented in various ways. The most frequently they are components of larger software systems. In this case, a typical approach relies on sorting the samples, as described above. However, with longer filter lengths or larger mask sizes (in case of image processing), median filtering may require relatively high computing power [5], with the computation complexity increasing with the square of the number of sorted samples.

In real time applications that additionally require low power data processing, e.g. in robotics or in automotive applications, a hardware realization may be a better option. Various solutions of this type have been reported in the literature. We can distinguish both mixed (analog-to-digital) [5], [6], [7], [8], [9] and purely digital [10] solutions. Selected examples of such realizations are briefly discussed below.

In one of the described approaches [5] the filter uses a binary search algorithm. The filter has been implemented in the CMOS 0.35 μm Floating-Gate-MOS technology. Thanks to the use of this technology the energy consumption has been substantially reduced. However, this was at the expense of larger chip area.

In [6] a filter working in the current mode, also realized in the CMOS 0.35 μm technology is presented. The filter is composed of current mirrors, current comparators and simple logic circuitry. The structure of the system resembles the structure of a binary tree. During the filtering, the signals are divided into groups (modules), each of them containing three signal samples. The number of such groups on the first layer of the tree equals the total number of inputs, n, divided by 3. The output signal, which a median value in a given group, is provided to the inputs a similar group at the following layer in the tree, etc. The structure of the circuit is relatively complex. According to the authors of the work, after a small expansion the system would enable sorting operation.

Another median filter working in the current mode, implemented in the CMOS 0.8 μm technology, has been described in [7]. The filter consists of three main parts, with one of them being a block that calculates the minimum value. The system works fully asynchronously. Although the solution is interesting, a disadvantage is that it allows filtering a set containing only 3 signal samples. In case of expanding the

978-1-5386-6768-2/18/$31.00 ©2018 IEEE

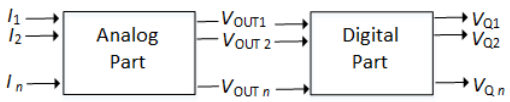

Fig. 1. A general block diagram of the proposed median filter.

circuit so that to enable the filtering over a longer data set, its structure complexity will be much larger. A similar solution was previously shown in [8].

In [9] a median filter is presented, which consists of two blocks: analog delay cells using the first-order all-pass circuits and transconductance comparators. The reported delay of the analog part is in-between 100 and 350 ns, for the input currents in the range from 40 to 80 μA.

In addition to described analog solutions, we can also indicate fully digital solutions. For example, in [10] a fully digital median filter is presented, realized in the CMOS 0.8 μm process. The filter is based on a novel bit-level running algorithm with a modular and parallel in structure. The system operates using a bit-level pipeline architecture, sorting signals from their most significant bits (MSBs) to the least significant bits (LSBs).

Considering the described state-of-the-art solutions, our proposed filter belongs to the mixed analog-digital group. The advantage of the proposed solution is that it can be easily adapted to calculating different quantiles, among which the median is only one of the possibilities. It is also a system whose hardware complexity increases linearly with the number of filtered signals. Any number of the inputs may be used.

II. Proposed Median Filter

A general block diagram of te proposed median filter is shown in Fig. 1. The filter is composed of analog and digital modules, shown in Figs. 2 and 3, respectively. The analog part asynchronously activates the corresponding elements in the digital block, whose role is to indicate the address of the current being the median.

A. Analog Part of the Filter

The analog part consists of two subcircuits, namely a current-to-time converter (ITC) and a resistive voltage divider (RVD). After the reset of all ICTs, particular input currents,I_i, charge capacitors C throughout PMOS current mirrors (CM). The charging times are inversely proportional to the values of the input currents. When in a given ITC the voltage on the capacitor exceeds half of the supply voltage, the NOT gates at the ITC output are switched over. The differences in time moments when particular logic '1' signals appear at the ICTs outputs depend on the mutual relations between the values of the input currents.

The second block (RVD) of the analog part is composed of interconnected resistive sections and switches, as shown in Fig. 2(b). The operation of this block is based on a voltage divider, in which the output voltage is expressed as follows:

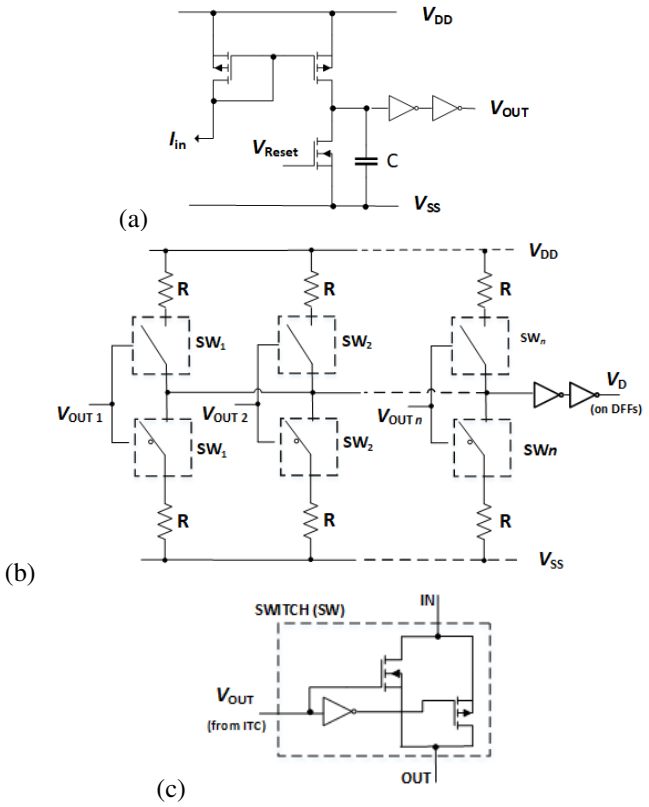

Fig. 2. Analog part of the filter: (a) Current to time converter (ITC), (b) Resistive voltage divider (RVD), (c) switches realized as transmission gates.

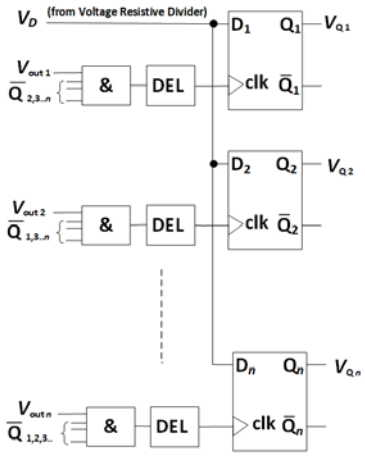

Fig. 3. Asynchronous digital control block of the filter.

$$V_{\mathrm{OUT}} = \frac{R_{\mathrm{N}}}{R_{\mathrm{P}} + R_{\mathrm{N}}} \cdot (V_{\mathrm{DD}} - V_{\mathrm{SS}})\,. \tag{1}$$

After replacing the R_{N} and R_{P} resistances with the G_{N} and G_{P} conductances, respectively:

$$G_{\mathrm{N}} = \frac{1}{R_{\mathrm{N}}} \quad \text{and} \quad G_{\mathrm{P}} = \frac{1}{R_{\mathrm{P}}}. \tag{2}$$

The conductance G_{P} between the V_{DD} and the V_{OUT} point in the circuit shown in Fig. 2(b), can be described as follows:

$$G_{\mathrm{P}} = S_{\mathrm{BIT1}} \cdot \frac{1}{R_1} + S_{\mathrm{BIT2}} \cdot \frac{1}{R_2} + S_{\mathrm{BIT3}} \cdot \frac{1}{R_3} + \cdots + S_{\mathrm{BIT}n} \cdot \frac{1}{R_n}. \quad (3)$$

Similarly the G_{N} conductance between the V_{SS} and the V_{OUT} point is given as follows:

$$G_{\mathrm{N}} = \overline{S_{\mathrm{BIT1}}} \cdot \frac{1}{R_1} + \overline{S_{\mathrm{BIT2}}} \cdot \frac{1}{R_2} + \overline{S_{\mathrm{BIT3}}} \cdot \frac{1}{R_3} + \cdots + \overline{S_{\mathrm{BIT}n}} \cdot \frac{1}{R_n}. \quad (4)$$

Finally, the V_{OUT} voltage can be expressed as follows:

$$V_{\mathrm{OUT}} = \frac{G_{\mathrm{P}}}{G_{\mathrm{N}} + G_{\mathrm{P}}} \cdot (V_{\mathrm{DD}} - V_{\mathrm{SS}}). \quad (5)$$

Note that, when all the input signals $S_{\mathrm{BIT}i}$ are equal to '1', the output voltage V_{OUT} approximately equals V_{DD}. On the other hand, when all the input signals $S_{\mathrm{BIT}i}$ equal '0', the V_{OUT} voltage equals V_{SS}.

Logical outputs signals from particular ITC blocks control corresponding resistive sections in the described RVD. At the beginning of a given calculation cycle (after reset of all capacitors), when all ITCs outputs are '0', the voltage at the RVD output is 0. When successive input currents cause switching over the NOT gates in corresponding ITCs, their resultant output '1' signals switch over particular resistance sections in the RVD. As a result, the voltage at the RVD output gradually increases. Regardless of the number of the resistive sections, always the median value of a set of the input currents causes the voltage at the RVD output to exceed the half of the supply voltage. At this moment, the NOT gates at the RVD output are switched over. In this way, the RVD signals the appearance of a median.

B. Digital Part of the Filter

The logic signal from the RVD is then processed by the digital part of the filter. In the digital block, shown in Fig. 3, particular D-flip flops (DFFs) correspond to particular ITCs. The logical output signal from the RVD is fed to D inputs of all DFFs. At the same time, particular digital outputs from the ITCs are fed to particular clk inputs of the DFFs, after an appropriate delays intentionally introduced by the delay (DEL) blocks. The role of the DEL circuits is to compensate a delay introduced by the RVD. In this way, the signals from the ITCs register a current state at D inputs of the DFFs.

Before the median value has been reached, all clk signals from the ITCs register '0' value in their DFFs. After reaching the median, all subsequent DFFs would record the values of '1'. To avoid this situation and to remember '1' only for the median current, an appropriate mechanism cuts off the remaining ITC outputs.

As shown in the next section, this mechanism works effectively for a different number of input currents. However,

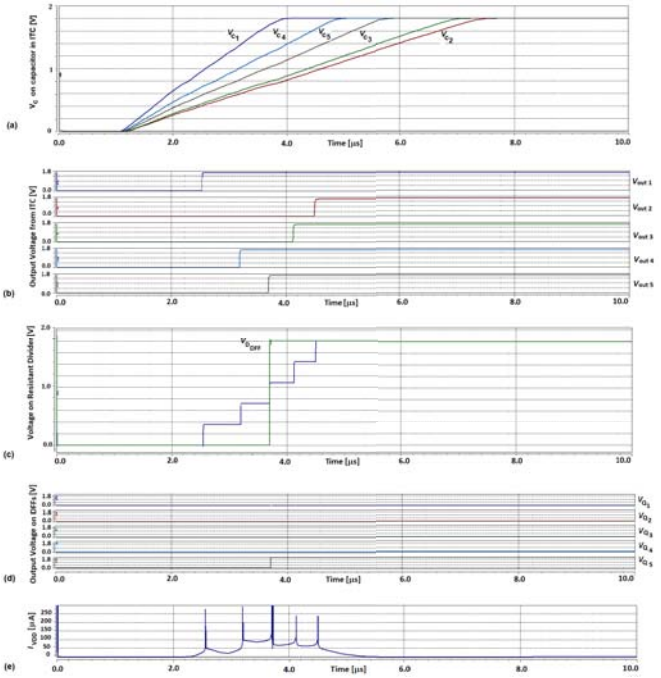

Fig. 4. Selected simulation results for the filter with five inputs.

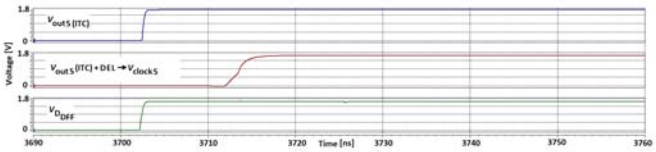

Fig. 5. Time dependencies between particular digital signals, illustrating necessary delays introduced by the DEL block to compensate delay of the RVD block, for the median filter with 5 inputs.

a certain problem may be due to inaccuracies in the circuit that may impact the assumed delays. Simulations for different values of the supply voltages and temperatures show that these differences are small and do not reduce the performance of the overl filter. The problem can be seen when several input currents have a value close to the median. In this situation, instead of the median, one of the currents close to it may be selected. In many applications of such filters, however, it is not a problem.

III. VERIFICATION OF THE PROPOSED FILTER

The proposed median filter has been realized in TSMC 180 nm CMOS technology and verified by comprehensive transistor-level simulations. To thoroughly verify the circuit, the simulations were carried out for various transistor models (slow, fast and typical), for different values of supply voltage (1 – 1.8 V) and different values of external temperature (-20 – 120°C) – PVT corner analysis.

In this section we present selected results for two exemplary median filters with 5 and 9 input signals (Figs. 4 and 6 respectively). The (a) diagrams in both figures show the

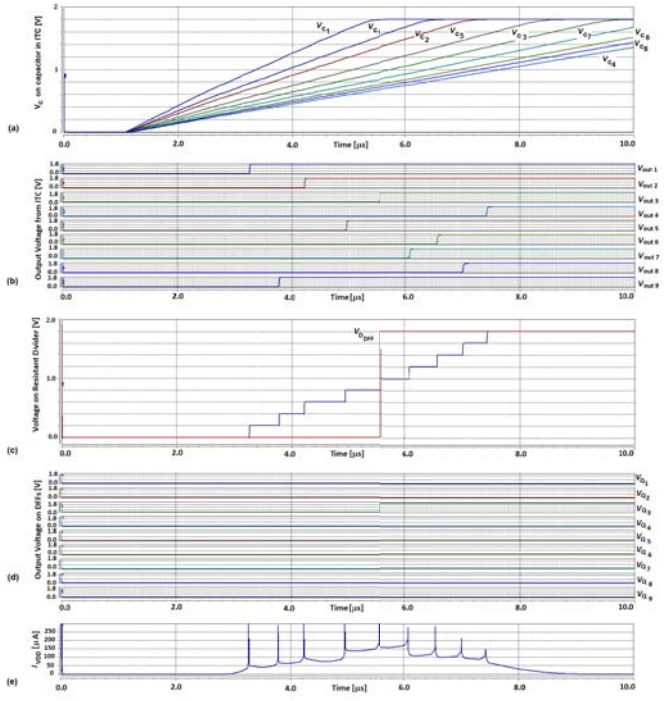

Fig. 6. Selected simulation results for the filter with nine inputs.

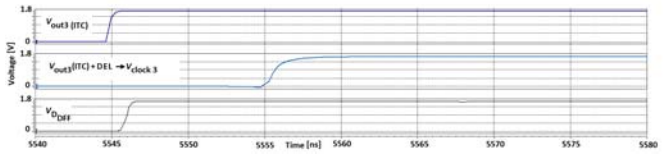

Fig. 7. Time dependencies between particular digital signals, illustrating necessary delays introduced by the DEL block to compensate delay of the RVD block, for the median filter with 9 inputs.

voltage increasing across the C capacitors in the ICTs due to values of the input currents. In this particular test, the following values of the currents have been selected: (i) for the version with 5 inputs I_i = 30, 13, 14, 22, 18 nA, (ii) for the version with 9 inputs I_i = 45, 32, 23, 16, 26, 18, 20, 17, 37 nA. Diagrams (b) show the switching moments of the outputs of particular ITC blocks. Diagrams (c) illustrate how the voltage rises at the analog output of the RVD, as well as the moment of switching over the digital output of the RVD. The next panel shows the outputs from the DFFs. As can be seen, only one of these outputs is activated, indicating the median value. Diagrams (e) show the supply current. Note that the maximum current flows during the switching process of the digital part.

Figs. 5 and 7 illustrate time dependences between the signals provided to the DFFs. It is important to assure that the clock impulse at the clk input of a given DFF appears a little bit later after the impulse at the D input of the DFF. To make it possible we used additional block that delay the clk impulses. As can be seen in Figs. 5 and 7 the clock signal in relation to the signal at the D input of the trigger is delayed approximately by 10 ns. This tie is sufficient to compensate the delay introduced by the analog part of the filter. We observe a similar dependency for different PVT parameters, described above, so the filter works properly.

Theoretically, the possible impact of the lack of a good described synchronization may result in the fact that instead of the median value, a little different value will be registered as an output signal. However, in most filtering tasks it is not a significant problem.

IV. CONCLUSIONS

In this work we presented a novel approach to hardware implementation of nonlinear filters. The problem with the median filters is the necessity of sorting signal samples, whose complexity increases with the median filter length. In case of software realizations, the problem basically does not exist, as sorting can be easily adapted to any number of samples. Additionally various sorting algorithms are available. The situation is different in hardware realizations, where the number of samples directly impacts the structure of the circuit.

The solution proposed in this work can be easily adapted to different numbers of the input signals. To make it possible, particular resistive sections in the RVD block could be connected / disconnected from the analog output. In this case additional configuration switches would be required.

The proposed filter offers also some other possibilities. Currently, NOT gates are used at the RVD output, which means that the median value is determined. Instead of this gate a voltage comparator can be used. Depending on the value of the reference voltage connected to the comparator signals different from the median value may be determined (quantiles of different orders – median, quartiles, quintiles, etc.). What is important in this case, the filter structure would not have to be changed.

REFERENCES

[1] Yu-Chi Tsao, Ken Choi, "Hardware-efficient VLSI implementation for 3-parallel linear-phase FIR digital filter of odd length," IEEE International Symposium on Circuits and Systems, pp.998-1001, 2012.

[2] R. Długosz, K. Iniewski, "Power and area efficient circular-memory switched-capacitor FIR baseband filter for WCDMA/GSM," IEEE International Symposium on Circuits and Systems, pp.2326-2329, 2008.

[3] Jinsoo Cho, Jongwoo Bae, "Edge-adaptive local min/max nonlinear filter-based shoot suppression," IEEE Transactions on Consumer Electronics, Vol. 52, Iss. 3, pp.1107-1111, 2006.

[4] R. Długosz, A. Rydlewski, T. Talaśka, "Low power nonlinear min/max filters implemented in the CMOS technology," International Conference on Microelectronics (MIEL), pp.397-400, 2014.

[5] H. Yamasaki and T. Shibata, "A high-speed median filter VLSI using floating-gate-MOS-based low-power majority voting circuits," European Solid-State Circuits Conference (ESSCIRC), pp.125-128, 2005.

[6] Y. -C. Hung, S. -H. Shieh and C. -K. Tung, "A real-time current-mode CMOS analog median filtering cell for system-on-chip applications," IEEE Conference on Electron Devices and Solid-State Circuits, pp.361-364, 2007.

[7] C. Noisuwan, J. Nakasuwan, B. Knobnob, S. Chivapreecha and K. Dejhan, "A CMOS median filter circuit design," International Symposium on Communications and Information Technologies, pp.1089-1092, 2006.

[8] S. Vlassis and S. Siskos, "Current-mode analogue median filterm," IEEE International Conference on Electronics, Circuits and Systems, Vol. 3, pp.1353-1356, 1999.

[9] A. Diaz-Sanchez, Jaime Ramirez-Angulo, A. Lopez-Martin and E. Sanchez-Sinencio, "A fully parallel CMOS analog median filter," IEEE Transactions on Circuits and Systems II: Express Briefs, Vol. 51, Iss. 3, pp.116-123, 2004.

[10] Chun-Te Chen, Liang-Gee Chen and Jue-Hsuan Hsiao, "VLSI implementation of a selective median filter," IEEE Transactions on Consumer Electronics, pp.33-42, 1996.