Efficient Sensor Signal Filtering for Autonomous Wireless Nodes

Alessandro Depari, *Member, IEEE*, Alessandra Flammini, *Member, IEEE*, Daniele Marioli, Mauro Serpelloni, Emiliano Sisinni, *Member, IEEE*, and Andrea Taroni

***Abstract*—Wireless sensor networks are nowadays a reality. However, a wireless node must be a truly autonomous system, i.e., it must embed its own power source, which is generally a battery pack. For this reason, it is of main concern to limit every source of wasted power. In particular, nodes exploit low-duty-cycle strategies to increase their life, spending most of their time in low-power mode, turning off all devices within the system except a wake-up oscillator. However, this way, some time must be spent waiting for the circuitry settling time, particularly if a high-accuracy measurement is required. In this paper, an efficient implementation of a filtering algorithm that is able to reduce this time is discussed. It is based on a hybrid combination of a median and a mean filter, joining the advantages of both linear and nonlinear filtering. A well-tailored implementation has experimentally been tested and discussed. Two wireless prototypes have been realized to test the proposed filter performance for temperature and humidity measurements. The experimental results permit comparing the implemented filters in terms of computational time, power consumption, and the performance of noise filtering. For both sensors, the new filtering strategy resulted to be more efficient in terms of noise removal and consumption than traditional algorithms. In particular, in the case of slow sensors such as a humidity one, delays are dictated by the sensor itself, and computational time can be neglected, while for fast sensors such as a temperature one, the proposed schema greatly improves the system update rate.**

***Index Terms*—Low power, sensor signal processing, wireless sensor.**

I. INTRODUCTION

WIRELESS sensor use is growing in many application fields such as home automation, environmental monitoring, and process control [1]. Wireless transceiver cost is quickly decreasing, and emerging technologies (e.g., IEEE802.15.4) allow power reduction to obtain autonomous and mobile sensors [2]. Nowadays, batteries are the most common way to realize autonomous sensors, and several research activities concern with the efficient use of batteries [3]. On the other hand, many methods of power harvesting (solar cell, electromagnetic fields, piezoelectric generators, thermopiles, etc.) are growing [4]; they can be used to recharge batteries or to replace batteries. In both cases, low-power techniques must be adopted to save the small amount of available power. In most of applications, the wireless sensor periodically wakes up, measures the interesting quantity, and, if necessary, transmits information. In the case of a humidity or temperature sensor, information transmission rarely occurs, because humidity or temperature variation is very slow. In most cases, the measure is repeated every second, while data transmission takes place less than one time every minute. For this reason, to save further power, the sensing element and the interface circuits are powered just before the measurement, and then, they are turned off as soon as the measurement is over. Both the sensing element and the electronic interface circuits require a certain time to furnish a stable and accurate measurement.

In the following, a temperature/humidity sensor is taken into account, but the proposed approach can be used with other types of sensors. Typical temperature or humidity sensors used in low-cost applications are resistive or capacitive sensors. In the first case, the sensor can be inserted in a voltage divider, and the output voltage is acquired by means of an analog-to-digital (AD) converter. In the second case, the sensor is used in an oscillating circuit, e.g., based on the 555 multivibrator device, and a microcontroller estimates the output signal period by exploiting its timing unit. In both cases, the circuits must be operative for a short time, so that the power consumption could be limited. On the other hand, the use of filters is highly advisable to improve the signal-to-noise ratio. However, low-pass filters applied to the signals and power supply cause a long measure settling time. As a consequence, an enlargement of the operative time of the sensor, microcontroller, and interface circuits is produced, which determines an increment in the power consumption.

A simple and largely employed solution is to wait for a fixed time to perform the measure after the power supply has been applied to the sensors and the circuits. This fixed time value is established in the project stage by means of *a priori* considerations about the system. However, this solution is neither flexible nor efficient, because the time value is usually oversized. This is due to the circuit output settling time, which is not usually specified in the component data sheets, and furthermore, it strictly depends on the temperature, power supply, and load conditions. In addition, even for the same component, it can vary among different manufacturers, particularly if the component is not specifically developed for low-power applications.

For these reasons, usually, the output signal from the sensor is continuously measured until it reaches a stable value. This operation is not always simple, because of the noise overlapped to the signal, which makes the time to get a stable value

Manuscript received July 1, 2008; revised January 7, 2009. First published July 24, 2009; current version published December 9, 2009. The Associate Editor coordinating the review process was Dr. Domenico Grimaldi.

A. Depari, A. Flammini, D. Marioli, M. Serpelloni, and E. Sisinni are with the Department of Electronics for Automation, Faculty of Engineering, University of Brescia, 25123 Brescia, Italy (e-mail: mauro.serpelloni@ing.unibs.it).

A. Taroni is with the Università "Carlo Cattaneo" (LIUC), 21053 Castellanza, Italy.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIM.2009.2023104

longer. From a different point of view, this is a typical pattern recognition problem, made up of a preprocessing, a feature extraction, and a detection phase. Feature extraction can be performed by means of edge detection (analyzing the gradient of the monitored signal), while for the detection, a simple threshold rule can be used. However, these steps work well only in the absence of noise, which must be eliminated by the preprocessing stage. In conclusion, the use of a suitable filter is highly recommended; it should be effective toward different noise typologies, but it does not have to increase too much the measure time. In addition, it should be fast and easy to implement to not increase the microcontroller cost. In case of a signal affected by Gaussian noise overlapped with impulsive noise (IN), as it happens, for instance, on audio or echocardiographic signal analysis, the literature shows a large use of filters based on a combination of median, weighted-median, and linear filters for the sensor output smoothing [5], for example, the median hybrid filters suggested by Heinonen and Neuvo [6], [7], running medians with robust regression, which were analyzed by Davies *et al.* [8], and $Q\alpha$-method-based filters proposed by Croux and Rousseeuw [9], [10]. For audio signal elaboration and image processing, many realizations of efficient median filters have been proposed [11]–[14], even if usually, in this field, fast and powerful processors or DSPs are employed. Of particular interest are the so-called trimmed or winsorized mean filters, which are better described in the next section [15]. Generally, such filters are well designed for the signal of interest, but they are not easy to implement and need a lot of computational resources, requiring the microcontroller to be operative for a long time. In wireless autonomous devices, particular attention must be given to elaboration time and computational frequency. On the other hand, in the industrial field, the wireless sensors are subjected to different noises such as IN and Gaussian noise. As such, the algorithm implementation for autonomous wireless nodes can considerably be different from the traditional filters reported above: an adequately efficient implementation must consider short computational time, low power consumption, and effective noise filtering.

In a previous work [16], the development of a simple filter that is easy to implement and suitable to the measurement stabilization detection has been reported. In this paper, the filtering technique has been implemented in a more efficient way and tested on a real working case. The two wireless prototypes have been used to verify the proposed filter performance in comparison with that of traditional filtering strategies. The further experimental results permit comparing the implemented filters in terms of computational time, power consumption, and the performance of noise filtering.

II. Proposed Approach

As we said, the combination of a median and a mean filter is suitable for the application, but the implementation must carefully be tailored to be effective without requiring a lot of computational effort. The following analysis of the application and noise is needed to better characterize the filter requirements. The signal coming out from the sensor can be approximated as the summation of three terms: the first one represents the useful information, the second one models the broadband additive white Gaussian noise (AWGN), and the last one models IN. As is well known, broadband noise can effectively be removed, or at least attenuated, by means of a low-pass finite-impulse-response (FIR) filter (as the moving average), while IN can be removed using nonlinear filtering as median filters. Obviously, the span of the filter must be chosen to prevent underlying signal distortion, and it is regulated by a tradeoff. A filter with too many taps requires high computational effort and has a high latency (not suitable for fast sensors such as resistive ones). On the contrary, AWGN and glitch immunity cannot be ensured with too few taps. Considering that the sampling time could be in the range [10,100] μs for resistive sensors and in the range [0.1,1.0] ms for capacitive sensors, the overall elaboration time must be on the order of 100 μs per sample. As an example, Fig. 1 shows the output of a capacitive sensor used together with a TLC555 oscillator; the measurement readout is the signal period (time elapsed between two successive rising edges, which is on the order of 100 μs). The jitter, which is due to switching broadband noise, makes measurements T1, T4, and T5 different; on the contrary, the glitch makes T2 and T3 outlier data that must be discarded.

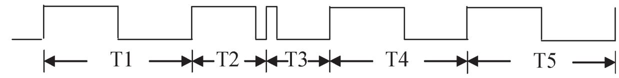

Fig. 1. Output of a capacitive sensor used with TLC555 affected by an outlier.

With regard to AWGN suppression, supposing that we implement a linear-phase FIR filter (i.e., the group delay is constant at all frequencies), an N-tap filter has a latency equal to $Ts(N-1)/2$ (in seconds), where Ts is the sampling period. Probably, the best choice is a moving average filter that provides the fastest step response for a given noise reduction.

Referring to IN, how many spikes can effectively be rejected? To answer this question, the estimator breakdown point is usually adopted; it is defined as the largest fraction of input data that can be replaced by arbitrarily large values without driving the estimator output error to infinity. The breakdown point of a median of $N = 2K + 1$ data points is K/N, i.e., at least half of the sample needs to be replaced to completely destroy the filter output. On the contrary, the mean filter has a breakdown point equal to zero, regardless of the filter length.

The aim of this paper is to explore efficient implementations of a median filter able to reject at least two outliers, in conjunction with a moving average filter with an overall delay on the order of twice the sampling time. In other words, the median filter must have a minimum length of five, and the moving average filter must be a four-tap filter providing a new output value every four input samples, still halving the noise. However, even if the moving average is an exceptionally good smoothing filter (the action in the time domain), it is an exceptionally bad low-pass filter (the action in the frequency domain). In addition, the nonoverlapping windowing of the median filter leads to a decimation in time, worsening spectral properties [Fig. 2(a)]; in fact, the cutoff frequency is equal to $(Ts \cdot N \cdot M) - 1$, where Ts is the sampling period, N is the median filter length, and M is the moving average length, respectively. To overcome this limit, it is possible to use overlapping windows shifted

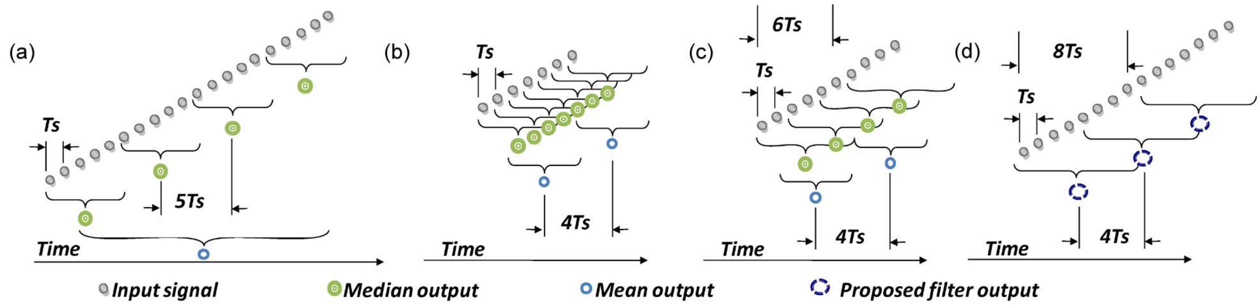

Fig. 2. Comparison of (a) the traditional nonoverlapping approach, (b) and (c) modified overlapped hybrid median–mean filters, and (d) the proposed eight-tap filter.

by one point if an odd length is chosen and two points if an even length is adopted. For instance, an overlapped five-tap median filter followed by a four-tap moving average filter [Fig. 2(b)] has an absolute better noise bandwidth with respect to the nonoverlapped one [Fig. 2(a)]. A six-tap median filter, which updates its output, i.e., the average of the two central points, every two input samples, followed by a two-tap moving average filter [Fig. 2(c)] has about the same time properties as the previous one [Fig. 2(b)]. In fact, both solutions are able to filter out two outlier data (glitches with duration equal to two samples) and show the same behavior, with respect to a step and a monotonic input, as a four-tap moving average. Obviously, a nonoverlapping strategy does not satisfy our requirements; with regard to overlapping implementations, it can be shown that an even-length median followed by a two-tap average filter [Fig. 2(c)] has better frequency rejection properties than an odd-length median [Fig. 2(b)], due to different interleaving of linear and nonlinear filtering actions.

A further improvement can be obtained using the so-called truncated, trimmed, or winsorized mean [15]. In an ordered set X_i, the placement of the sample is referred to as the rank. Thus, in a set of cardinality $N = 2K + 1$, the median has rank $= K$, and the truncated mean filter (TMF) is given by the average of all the samples having rank $[K + 1 - Q, K + 1 + Q]$, where Q is a constant value fixed *a priori*. More formally

$$\mathrm{TMF}_{\mathrm{ODD}}(X_1, \ldots, X_N; Q) = \frac{\sum_{i=1}^{N} a_i X_i}{\sum_{i=1}^{N} a_i},$$

$$a_i = \begin{cases} 1, & \text{if } K + 1 - Q \leq \mathrm{rank}(X_i) \leq K + 1 + Q \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

If we considered an even length $N = 2K$, the previous equation must slightly be modified since the median is defined as the average between samples of rank K and $K + 1$, and (1) becomes

$$\mathrm{TMF}_{\mathrm{EVEN}}(X_1, \ldots, X_N; Q) = \frac{\sum_{i=1}^{N} a_i X_i}{\sum_{i=1}^{N} a_i},$$

$$a_i = \begin{cases} 1, & \text{if } K - Q \leq \mathrm{rank}(X_i) \leq K + 1 + Q \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

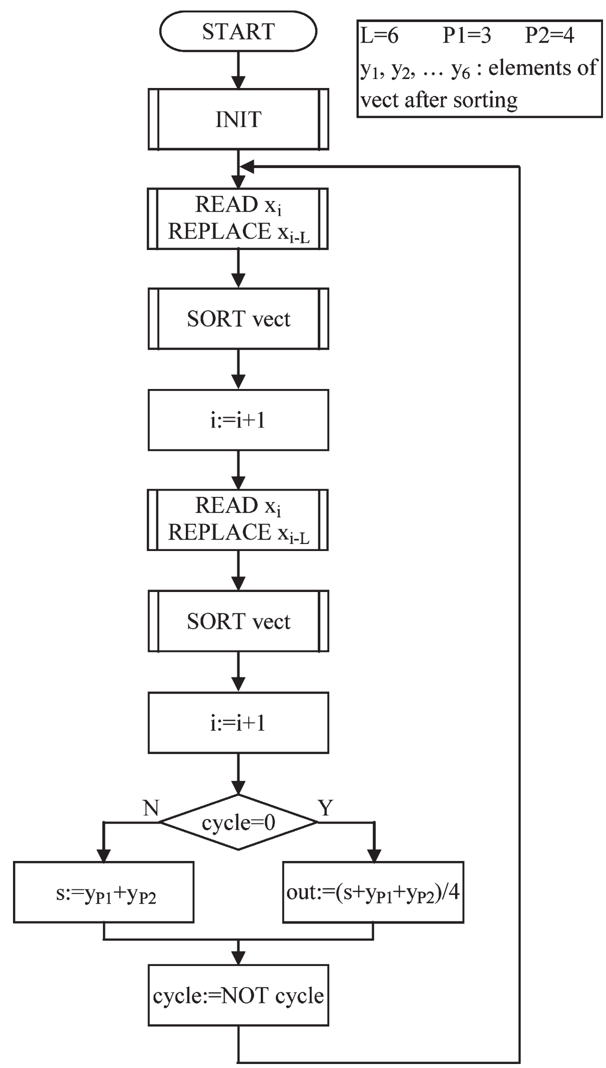

Fig. 3. Flowchart of the reference six-tap filter.

Following this reasoning, an eight-tap modified median filter [Fig. 2(d)], which is updated every four input samples and furnishes the average value of the four central points, exhibits an even steeper frequency response, still preserving a good behavior with respect to the step impulse, monotonic signals, and glitches.

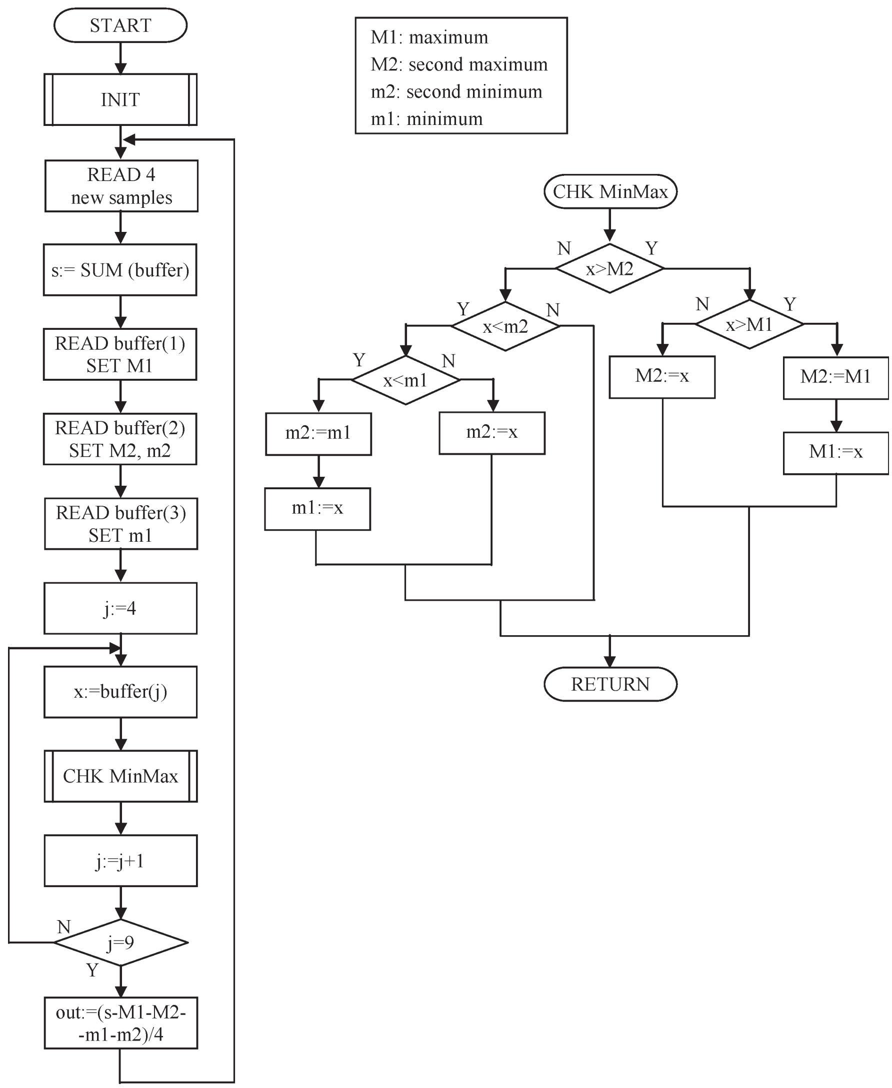

Fig. 4. Flowchart of the proposed eight-tap filter.

III. Filter Implementation

In the following, a new implementation of an eight-tap modified median filter will be described; an experimental evaluation will be carried out with real wireless sensors. As a reference, the above-described six-tap filter will be also tested [see Fig. 2(c)]. In Fig. 3, the flowchart of the reference six-tap filter is shown. We start from a six-element input vector; each element is a couple {value, time} representing the sample value and sampling instant, with $1 \leq$ time ≤ 6. Neglecting the initialization phase, a list structure has been implemented, and elements {value, time} are updated in the vector (READ X_i and REPLACE X_{i-6}) according to the bubble sort algorithm. As the vector is already ordered, the bubble sort for the new entry is a very efficient method (SORT vect). Every two updates, the median filter furnishes the average of the two central points. The overall system output is the average of two consecutive outputs of the median filter.

In Fig. 4, the flowchart of the proposed eight-tap filter is shown. It must be noticed that this implementation is different from the previous one that exploits the bubble sort algorithm; the proposed algorithm is focused on the discard of extremes, which are labeled $m1$ (minimum value), $m2$ (second minimum), $M2$ (second maximum), and $M1$ (maximum value) (with $m1 \leq m2 \leq M2 \leq M1$).

The basic idea is to partition the data array; starting from an initial sorted array, every new element is forced in the outlier group or in the correct sample one. This is the fastest way to proceed if no *a priori* knowledge of the signal is available [17]. In fact, in this application, we want to select outlier elements and not to sort them, i.e., we can attend only the subset that contains them. After that, we can perform averaging of correct samples.

First, three samples are sorted by a simple algorithm that initializes the $m1$, $m2$, $M1$, and $M2$ values. The remaining five samples are checked to replace $m1$, $m2$, $M1$, and $M2$. At the end of this process, the output result coincides with the mean value of the four central values (summation of all elements without $m1$, $m2$, $M1$, and $M2$).

Due to this approach, it is possible to decrease the computational effort, leading to an almost equal time for both the best and the worst case. The flowchart in Fig. 4 is a simplified one. In fact, samples are processed while they are acquired in a continuous and more efficient fashion.

IV. Experimental Setup

Two wireless prototypes have been developed to test the filters' performance: a temperature sensor (Pt100) with an interface circuit that provides an output voltage and a capacitive humidity sensor with a frequency-coded output signal. Both prototypes are supposed to be battery powered. To optimize the battery power (two AA NiMH rechargeable batteries—2400 mAh), a step-up dc/dc converter (Texas TPS61070) is necessary between the power source and the circuits.

The battery life L (in hours) of a wireless sensor depends on the battery capacity C (in ampere hours), the power supply efficiency η, the power supply output voltage gain Kv, and the total mean current consumption $I_{\mathrm{cc,mean}}$ (in amperes) according to $L = (\eta \cdot C)/(Kv \cdot I_{\mathrm{cc,mean}})$. Normally, the device wakes up every T_m seconds, takes about T_a (active phase) to start up and measure quantities, and transmits and receives information by the RF link every T seconds ($T > Tm$), taking time T_{RF}. $I_{\mathrm{cc,mean}}$ is shown in the following equation, where I_a, I_{RF}, and I_{sleep} are the whole-circuit current consumptions in the active, RF, and sleeping phases, respectively:

$$\begin{aligned} I_{\mathrm{cc,mean}} &= I_a \frac{T_a}{T_m} + I_{\mathrm{RF}} \frac{T_{\mathrm{RF}}}{T} \\ &\quad + I_{\mathrm{sleep}} \frac{T_m \cdot T - T_m \cdot T_{\mathrm{RF}} - T \cdot T_a}{T \cdot T_m} \\ &\approx I_{\mathrm{sleep}} + (I_a - I_{\mathrm{sleep}}) \frac{T_a}{T_m}. \end{aligned} \tag{3}$$

As in the case $T_{\mathrm{RF}} \ll T$, the current consumption $I_{\mathrm{cc,mean}}$ mainly depends on T_a. Therefore, as we said, T_a must sufficiently be high to guarantee a stable measurement but not too high to save power.

Both the realized prototypes use a low-power microcontroller (Freescale MC9S08GT60A) and a low-power IEEE802.15.4 transceiver (Freescale MC13192). The conditioning circuits must be simple to abate power consumption, and it should be possible to virtually turn off this circuitry without affecting the transient response.

The first device is a wireless resistance temperature detector (RTD) that uses a Pt100 as the sensible element. Fig. 5 shows the conditioning circuitry, while the following equation shows its input/output relation:

$$V\mathrm{out} = R_{\mathrm{Pt100}} \cdot \frac{V\mathrm{Ref}}{R3} \left(1 + \frac{R7}{R6}\right) \mathrm{mA} = R_{\mathrm{Pt100}} \cdot 26 \ \mathrm{mA}. \tag{4}$$

The sensor is driven by constant current to reduce the energy lost in the resistance of the wires. The current generator circuit, composed by both operational amplifiers U1 and U2, excites the sensor. An operational amplifier, i.e., A4, is used to zero wire resistance error. A fourth amplifier (U3) is used to amplify the signal and filter possible alias interferences and wideband

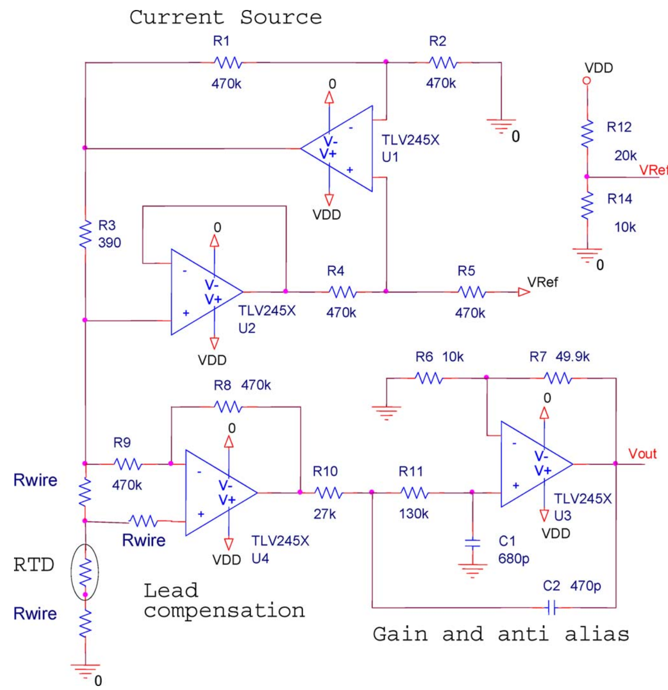

Fig. 5. Interface circuit of the Pt100.

noise. The 10-bit converter of the MC9S08GT60A converts the voltage across the RTD to digital code. Every amplifier is provided with a shutdown pin to enter in the low-power mode (Texas TLV2455). An example of the conditioning circuit output is depicted in Fig. 8; the sampling time is $Ts = 16$ μs, even if the effective rate is imposed by the filter computation time; and the cutoff frequency of the second-order Butterworth low-pass filter is about 5 kHz.

The second wireless device uses a capacitive sensor (Humirel HS1100) to measure relative humidity (RH). The signal conditioning circuit (Fig. 6) converts capacitance variations into a frequency-coded signal integrated circuit (IC) according to

$$f = \frac{1}{C@\%\mathrm{RH} \cdot (R5 + 2 \cdot R6) \cdot \ln 2.} \tag{5}$$

The variable capacitor U1 is connected to the TRIG and THRES pin of a timer 555. Pin 7 is used as a short circuit pin for resistor $R5$. The sensor capacitance is charged through $R6$ and $R5$ to the threshold voltage (approximately 0.67 Vcc) and discharged through $R6$ only to the trigger level (approximately 0.33 Vcc) since $R6$ is shortened to ground by pin 7. To provide an output duty cycle close to 50%, $R5$ should be very low compared to $R6$. The frequency output can be computed as depicted in Fig. 1(b). At Q, the sensor has a nominal capacity of $C = 180$ pF, so the conditioning circuitry gives a frequency of 7483 Hz (microcontroller input capture has a timing unit with 125 ns of resolution). According to the manufacturer, the average span of the capacity in the range $0\% < \mathrm{RH} < 100\%$ is 160 pF $< C(\mathrm{RH}) <$ 210 pF; this means that the minimum time interval for sample updating is on the order of 110 μs. The timer TLC555 lacks of a shutdown pin, so it is powered by a microcontroller output port (P2).

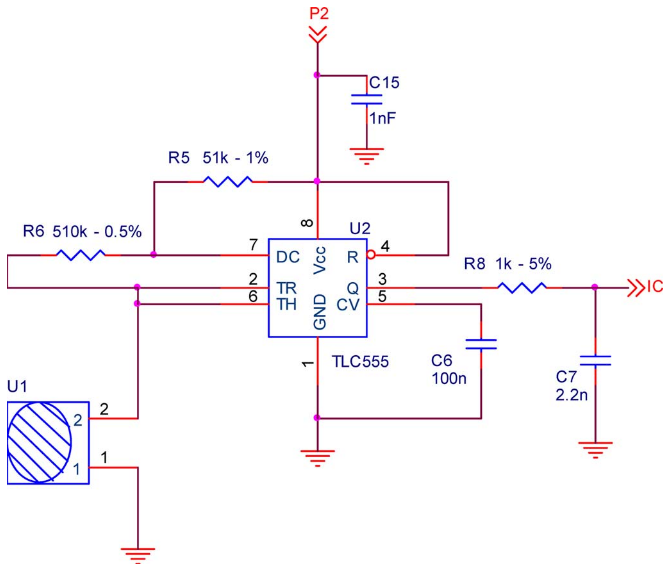

Fig. 6. Interface circuit of the capacitive sensor.

V. Experimental Results

The two wireless prototypes have been used to test the proposed filter performance in comparison with that of the reference and a traditional mean filter (moving average). In the following, we simply address the active (or measurement) phase of the wireless sensor, ignoring what happens in the sleep and transmit/receive phases.

The throughputs of the filters have been measured by exploiting two digital lines offered by the microcontroller. The first one is referred to the microcontroller activity, i.e., the signal is low during the computational activity, while the high level corresponds to inactivity time (which the microcontroller can exploit to perform other pending tasks or to switch in the sleep mode). The toggle from high to low occurs in correspondence of measurement data saving. The other digital line refers to the filter output data update: output data are saved into the microcontroller memory at every edge. It is important to underline that the measured time represents an upper bound; in fact, the filter algorithms have been implemented in plain ANSI-C, and no compiler optimizations have been set to be more hardware implementation independent.

In Fig. 7, the RH sensor signal and the digital line status are reported for the implemented filter. In Fig. 7(a), the activity time of the reference filter is longer than the sensor signal period; this means that the reference filter is not able to generate a coherent sample at every period of the sensor signal, since the data buffer is not correctly updated. On the contrary, the activity time of the proposed filter is shorter than the sensor signal period, thus ensuring a correct data output. In Fig. 7(b), a capacitor of 220 pF has been added to the humidity sensor to artificially increase the sensor signal period. In this configuration, the reference filter can also provide sampling data at every period of the sensor signal.

Table I reports and compares the duty-cycle values of the activity signals for both the reference and the proposed filter; three operative cases have been considered: sensor capacitance alone (C) and in parallel with two different capacitors ($C + 220$ pF and $C + 470$ pF). The table shows that the proposed filter correctly works for all the conditions, as previously affirmed, and always has a higher duty cycle than the reference filter, i.e., it is more computationally efficient.

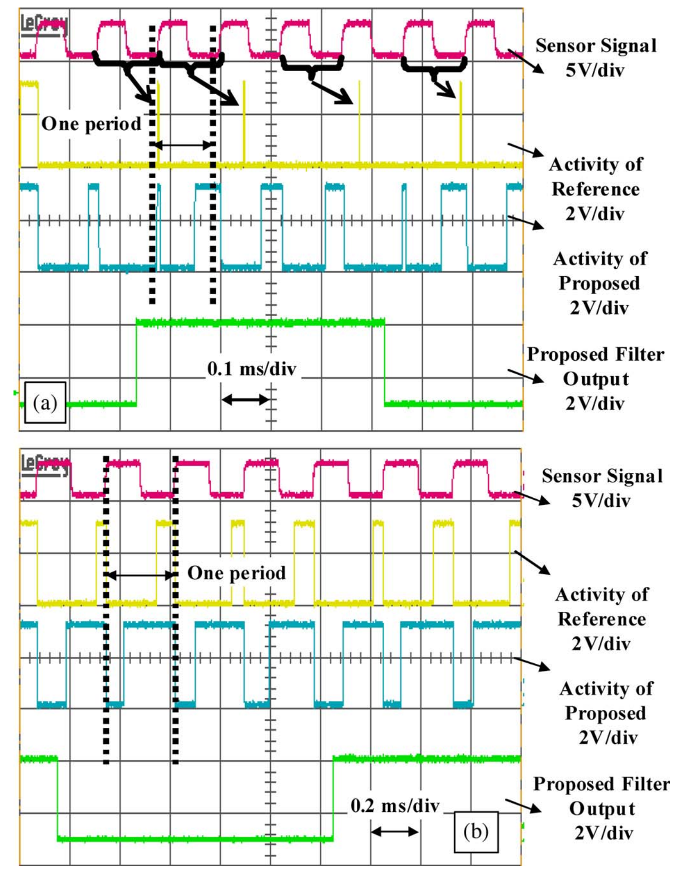

Fig. 7. Diagrams of the reference voltages for humidity measurements. (a) Reference versus proposed filter for a typical sensor signal. (b) Sensor + 220 pF.

TABLE I
Values of Duty Cycle of Activity Diagrams for the Two Filters for Different Operative Setups

Filter	Sensor	Sensor + 220 pF	Sensor + 470 pF
Reference	Under sampling	21.8 %	52.4 %
Proposed	25.6 %	66.8 %	80.7 %

The power consumption of the autonomous wireless node was kept to a minimum to maximize the lifetime of the batteries. As we said, the sensor normally sleeps, using low-power modes for both the microcontroller and the transceiver. In this phase (inactive), the power consumption is negligible. During the measuring phase (active and high-power consumption), the microcontroller is turned on, while the transceiver is turned on only if communication is required.

In Fig. 8, the diagrams of the microcontroller digital line status for temperature measurements are reported. In Fig. 8(a), the proposed filter is compared to a mean filter with four taps. The activity diagram of the mean filter presents a computational time lower than the proposed one; obviously, the algorithm complexity of the mean filter is lower as well. Due to this aspect, the output diagram of the mean filter has a period lower

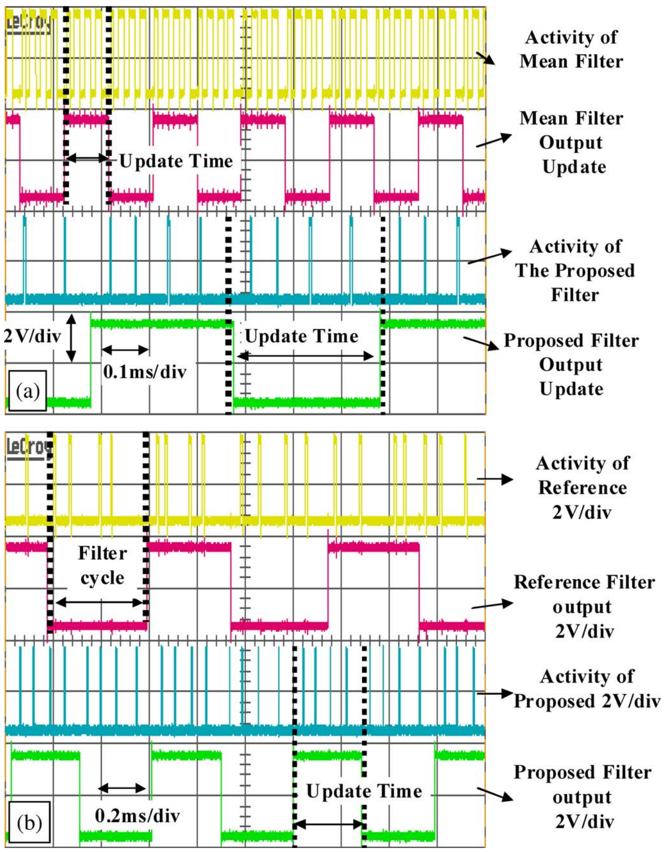

Fig. 8. Wireless transmission for temperature measurements for (a) mean filter versus eight-tap filter and (b) eight-tap filter versus six-tap filter.

than proposed one. In Fig. 8(b), the proposed filter is compared to the reference filter. The activity diagram of the reference filter presents a not-fixed computational time that depends on the bubble sort time. However, the output data period of the reference filter is approximately constant. Moreover, the filter cycle of the reference is higher than the one of the proposed filter.

In Table II, the update rate of the filter output for the implemented filters is reported as obtained by averaging 30 consecutive filter outputs. Temporal results obtained for this sensor can also be used as a lower bound for the capacitive sensor, which does not have a constant update rate, since the quantity of interest is frequency coded. As previously observed, the output data are saved into memory at every edge of the so-called "Filter Output" signal, so during a period of the output signal, two consecutive values are saved. The mean filter with four taps has the lowest period time, while the reference filter has the highest. The overall observation time of the proposed filter is in the middle, comparable with that of a mean filter with 16 taps.

For temperature measurements, the proposed filter represents a good tradeoff between the reference and the mean filter. The reference, compared with the others, has high computational time, which means a long activity time, but ensures an adequate noise removal. On the contrary, the mean filter has a lower computational time, which means a shorter activity time, but the performance of IN filtering is worst. It should be noticed that a good method to reduce microcontroller power consumption

TABLE II
UPDATE RATE OF THE IMPLEMENTED FILTERS FOR TEMPERATURE MEASUREMENTS

Filter	Update Time
Mean (4-taps)	92 μs
Mean (16-taps)	284 μs
Proposed (Hybrid 8-taps)	296 μs
Reference (Hybrid 6-taps)	387 μs

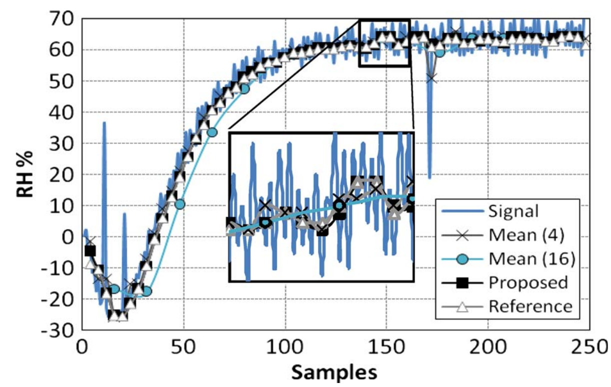

Fig. 9. Recorded humidity data for a sensor signal with noise for the implemented filters.

is to slow down its clock frequency; this way, the time taken by the mean filter could equate the one taken by the reference; consequently, an overall power reduction could be achieved. Therefore, the proposed filter can be attractive for wireless application in which low-power consumption and good noise-filtering performance are important characteristics.

The temperature measurements represent a time boundary due to a very small settling time of the sensor signal. In fact, even if the AD sampling time is imposed by the hardware ($\mathrm{Ts} = 16\ \mu\mathrm{s}$), the effective update rate (which determines the overall system bandwidth) depends only on the computation time. For this reason, the time-filtering performance of the implemented filter has been tested for temperature measurements and is also valid for the humidity sensor.

In Fig. 9, the implemented filters have been tested with a real-world noisy signal. The same signal has been used for each filter. The signal presents a white (broadband) noise and a spike (impulsive) noise. As expected, the proposed and the reference filter have the same behavior, while the mean (both the four- and eight-tap) filters have the worst behavior with regard to the IN (they are not able to eliminate the spike).

In summary, the proposed filter has comparable filtering characteristics to those of the reference filter, i.e., it is able to reject both AWGN and IN but, as reported, with a lower power consumption.

VI. CONCLUSION

In this paper, an efficient implementation of a hybrid median–mean filter has been described and tested. The implemented filter adopts efficient strategies that consider the wireless autonomous node as an application target. It has purposely been designed for a faster wake-up time of an autonomous wireless node. In this kind of applications, nodes spent most of their time in low-power mode, turning off all devices within the node, except a low-power oscillator. Obviously, this leads to a transient that must be discarded to obtain a good-quality readout. The developed filter is able to preserve the step response, still rejecting broadband noise. The experimental results for humidity measurements show that the proposed filter correctly works for all the sensor conditions and is more computationally efficient. For temperature measurements, the proposed filter represents a good tradeoff between the reference and the mean filter. The reference has a higher computational time, while the mean filter has worse filtering capability. Short computational time, low power consumption, and effective noise filtering are key parameters, as experimentally tested. Therefore, the proposed filter can be attractive for wireless application in which low power consumption and good noise-filtering performance are important characteristics. As suggested, it can advantageously be used to improve the transient behavior of low-"duty-cycle" autonomous sensors.

REFERENCES

[1] P. Baronti, P. Pillaia, V. W. C. Chooka, S. Chessab, A. Gottab, and Y. F. Hua, "Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards," *Comput. Commun.*, vol. 30, no. 7, pp. 1655–1695, May 2007.

[2] J. A. Gutiérrez, E. H. Callaway, Jr., and R. L. Barrett, Jr., "Low-rate wireless with IEEE 802.15.4," *IEEE Standards Wireless Networks Series*, pp. 3–12, 2004.

[3] A. Jossen, "Fundamentals of battery dynamics," *J. Power Sources*, vol. 154, no. 2, pp. 530–538, Mar. 2006.

[4] N. G. Stephen, "On energy harvesting from ambient vibration," *J. Sound Vib.*, vol. 293, no. 1/2, pp. 409–425, May 2006.

[5] U. Gather and R. Fried, "Methods and algorithms for robust filtering," in *Proc. COMPSTAT*, 2004, pp. 159–170.

[6] P. Heinonen and Y. Neuvo, "FIR-median hybrid filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 6, pp. 832–838, Jun. 1987.

[7] P. Heinonen and Y. Neuvo, "FIR-median hybrid filters with predictive FIR substructures," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 6, pp. 892–899, Jun. 1988.

[8] P. L. Davies, R. Fried, and U. Gather, "Robust signal extraction for on-line monitoring data," *J. Stat. Plan. Inference*, vol. 122, no. 1/2, pp. 65–78, May 2004.

[9] C. Croux and P. J. Rousseeuw, "Time-efficient algorithms for two highly robust estimators of scale," in *COMPSTAT*, 1992, pp. 411–428.

[10] P. J. Rousseuw and C. Croux, "Alternatives to the median absolute deviation," *J. Amer. Stat. Assoc.*, vol. 88, no. 424, pp. 1273–1283, 1993.

[11] O. Vainio, Y. Neuvo, and S. E. Butner, "A signal processor for median-based algorithms," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 9, pp. 1406–1414, Sep. 1989.

[12] C. Chandra, M. S. Moore, and S. K. Mitra, "An efficient method for the removal of impulse noise from speech and audio signals," in *Proc. IEEE ISCAS*, May 31–Jun. 3 1998, vol. 4, pp. 206–208.

[13] P. J. S. G. Ferreira, "Sorting continuous-time signals: Analog median and median-type filters," *IEEE Trans. Signal Process.*, vol. 49, no. 11, pp. 2734–2744, Nov. 2001.

[14] I. Kauppinen, "Methods for detecting impulsive noise in speech and audio signals," in *14th Int. Conf. DSP*, Jul. 1–3, 2002, vol. 2, pp. 967–970.

[15] J. Astola and P. Kuosmanen, *Fundamentals of Nonlinear Digital Filtering*. Boca Raton, FL: CRC Press, 1997.

[16] S. Bicelli, A. Depari, A. Flammini, D. Marioli, M. Serpelloni, E. Sisinni, and A. Taroni, "Sensor signal processing for autonomous wireless sensors," in *Proc. I2MTC*, 2008, pp. 1210–1215.

[17] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge, U.K.: Cambridge Univ. Press, 1992.

Alessandro Depari (S'07–M'08) was born in Breno, Italy, in 1976. He received the Laurea degree in electronic engineering and the Ph.D. degree in electronic instrumentation from the University of Brescia, Brescia, Italy, in 2002 and 2006, respectively.

Since 2007, he has been an Assistant Professor (Researcher) with the Department of Electronics for Automation, Faculty of Engineering, University of Brescia. His research interests include signal conditioning and processing for chemical sensors, particularly resonant and resistive sensors for electronic noses; the development of sensor networks for distributed measurement; and the design of methods and digital electronic circuits for numeric measurement instrumentation.

Alessandra Flammini (M'99) was born in Brescia, Italy, in 1960. She received the Laurea degree (with honors) in physics from the University of Rome, Rome, Italy, in 1985.

From 1985 to 1995, she worked on industrial research and development on digital drive control. From 1995 to 2002, she was a Researcher with the Department of Electronics for Automation, Faculty of Engineering, University of Brescia, Brescia, where she has been an Associate Professor since 2002. She teaches several courses about measurements in industrial environments, digital electronics, and microprocessor-based systems. Her main research activity is the design of methods and digital electronic circuits for numeric measurement instrumentation, sensor signal processing, smart sensor networking, and fieldbus applications.

Daniele Marioli was born in Brescia, Italy, on January 21, 1946. He received the Laurea degree in electrical engineering from the University of Pavia, Pavia, Italy, in 1969.

Since his graduation, he has been involved in research and educational activities with the Politecnico di Milano, Milan, Italy, and the University of Brescia, where he has been a Full Professor of electronics since 1990. He is the Chief of the Department of Electronics for Automation, Faculty of Engineering, University of Brescia. He is the author or a coauthor of more than 200 scientific papers published on international and national journals and conference proceedings. He is a coholder of four patents. His research interests include the design, realization and test of sensors, electronic instrumentation and signal processing electronic circuits. His activities in these fields are related to the realization of innovative sensors in thick-film technology, based on piezoelectric, piroelectric, and piezoresistive behaviors of screen printable pastes, and in MEMS technology, for the detection of physical quantities (acceleration, force, pressure, mass, etc.); the realization of high-resolution electronic instrumentation for capacitive measurements; the design and realization of integrated electronic circuit as a front end of piezoresistive-based sensors; the development of new linearization techniques based on neural networks; and the development of web sensors and wireless sensors.

Mauro Serpelloni was born in Brescia, Italy, in 1979. He received the Laurea degree (*summa cum laude*) in industrial management engineering and the Research Doctorate degree in electronic instrumentation from the University of Brescia, Brescia, in October 2003 and March 2007, respectively.

He is currently a Research Assistant of electrical and electronic measurements with the Department of Electronics for Automation, Faculty of Engineering, University of Brescia. He has worked on several projects relating to design, modeling and fabrication of measurement systems for industrial applications. His research interests particularly include contactless transmissions between sensors and electronics, contactless activation for resonant sensors, and signal processing for microelectromechanical systems.

Emiliano Sisinni (SM'02–M'05) was born in Lauria, Potenza, Italy, in 1975. He received the Laurea degree in electronics engineering and the Ph.D. degree in electronic instrumentation from the University of Brescia, Brescia, Italy, in 2000 and 2004, respectively.

Since 2005, he has been a Researcher (Assistant Professor) with the Department of Electronics for Automation, Faculty of Engineering, University of Brescia. His research activity focuses on numerical signal analysis, with particular interest in DSP-based instrumentation. He has been involved in the development of new NDT for ferromagnetic materials using the Barkhausen noise effect and the design of high-resolution low-cost instrumentation for motion estimation using inductive position sensors or optical encoders. Recently, he has been involved in the development of the department wireless sensor networking laboratory.

Andrea Taroni was born in 1942. He received the Laurea degree in physical science from the University of Bologna, Bologna, Italy, in 1966.

From 1971 to 1986, he was an Associate Professor with the University of Modena, Modena, Italy. From 1986 to 2007, he was a Full Professor of electrical measurements with the University of Brescia, Brescia, Italy. Since 2007, he has been the Chancellor of the Università "Carlo Cattaneo" (LIUC), Castellanza, Italy. He has done extensive research in the field of sensors for physical quantities and electronic instrumentation, developing both original devices and practical applications. He is the author of more than 100 scientific papers.