Metrol. Meas. Syst., Vol. XVII (2010), No. 2, pp. 217-232

PAN
POLSKA AKADEMIA NAUK

METROLOGY AND MEASUREMENT SYSTEMS

Index 330930, ISSN 0860-8229
www.metrology.pg.gda.pl

M&MS

FIXED-POINT IMPLEMENTATION OF INFINITE IMPULSE RESPONSE NOTCH FILTERS

Eduardo Pinheiro[1], **Octavian Postolache**[2], **Pedro Girão**[1]

1) Instituto de Telecomunicações and Instituto Superior Técnico, Av. Rovisco Pais, 1, 1049-001 Lisbon, Portugal
(✉ eduardo.pinheiro@lx.it.pt, p.girão@lx.it.pt, +351 218 418 454*)*
2) Instituto de Telecomunicações and Instituto Politécnico de Setúbal, Campus do IPS, Estefanilha, 2910-761 Setubal, Portugal
(opostolache@lx.it.pt)

Abstract

Many studies have been developed aiming to improve digital filters realizations, recurring to intricate structures and analyzing probabilistically the error´s behavior. The work presented in this paper analyzes the feasibility of fixed-point implementation of classical infinite impulse response notch filters: Butterworth, Chebyshev I and II, and elliptic. To scrutinize the deformations suffered for distinct design specifications, it is assessed: the effect of the quality factor and normalized cut-off frequency, in the number of significant bits necessary to represent the filter’s coefficients. The implications brought to FPGA implementation are also verified.

The work focuses especially on the implementation of power line notch filters used to improve the signal-to-noise ratio in biomedical signals. The results obtained, when quantizing the digital notch filters, show that by applying second-order sections decomposition, low-order digital filters may be designed using only part of double precision capabilities. High-order notch filters with harsh design constraints are implementable using double precision, but only in second-order sections. Thus, it is revealed that to optimize computation time in real-time applications, an optimal digital notch filter implementation platform should have variable arithmetic precision.

Considering these implementation constraints, utmost operation performance is finally estimated when implementing digital notch filters in Xilinx Virtex-5 field-programmable gate arrays. The influence of several design specifications, *e.g.* type, and order, in the filter’s behavior was evaluated, namely in regard to order, type, input and coefficient number of bits, quality factor and cut-off frequency. Finally the implications and potential applications of such results are discussed.

Keywords: digital filter implementation, digital filter word length effects, notch digital filters.

1. Introduction

Notch filters are very important in a wide variety of instrumentation applications, from telecommunications to biomedical signals processing, where often it is necessary to remove a narrow band or even a single frequency of the measurement signal. Digital implementation of these filters is preferable to an analog implementation due to drift absence and straightforward design of higher quality factors. Nevertheless, digital filter implementation has accuracy limitations due to the arithmetic’s finite precision [1–4], an issue that is much more significant in fixed-point arithmetic than in a floating-point one.

Due to the ease of designing and calculating the coefficients of high-performance digital finite and infinite impulse response (FIR and IIR) filters, the filter outcome is taken for granted, but, particularly if dealing with limited capacity fixed-point platforms (such as microcontrollers, digital signal processors, and field-programmable gate arrays) or with very demanding design constraints, the filtering stage may have a pernicious effect on the signal, completely missing its purpose.

Article history: received on Apr. 19, 2010; accepted on Jun. 8, 2010; available online on Jun. 16, 2010.

This problem has been studied [2–9] and, disregarding additional error sources originating from the A/D and D/A conversions, the three key error types are:

I. Quantization of the input signal into a finite set of discrete levels;
II. Representation of the filter coefficients by a short number of bits;
III. Propagation of rounding errors occurring in arithmetic operations.

To evaluate these errors influence in the final filter output, several approaches have been proposed [2–9, 12–14]. If input quantization errors, denoted above as type-I, are assumed to be random variables with a uniform probability distribution, a number of analysis tools is available to characterize their behaviour [10–14]. Errors of type-III are incessantly subject of reductions through the implementation of novel structure variations [1–2, 5, 15–17] based in state-space structures and direct form I with error feedback, also known as noise shaping or error spectrum shaping [5, 9, 18].

A short number of bits to represent the filter's coefficients, errors previously denoted as type-II, also have a comprehensive bibliography, reporting studies on important implementation issues. Some instability thresholds due to these errors were derived [6, 19], not including notch filters, coefficients sensitivity approach [15, 18–20], and structural changes to minimize the impact of these errors [2, 5, 8, 17], but none of these approaches considered notch filters.

Considering specifically biomedical applications, some studies have analyzed the digital filters distortion effect on the signal [15], but the feasibility and the outcome of the implementation has only recently been discussed [21]. Moreover, several biomedical studies ignore, to some extent, the higher-frequency components of the signals, implementing low-pass filters, or wide band-stop filters. Ballistocardiograms, electrocardiograms, electroretinograms which have sampling frequencies from 200 Hz to 2 kHz, and other biomedical signal high-resolution processing systems benefit from the use of power line notch filters. However, recent applications of biomedical systems tend to use wireless communications, limiting the maximum sampling frequency to about 200 Hz, so the fundamental power line frequency is the only concern. Hence notch filters are apposite to such applications, instead of comb filters, which may be of interest in higher sampling rates.

Since acquisition systems work at distinct sampling rates, the analysis of IIR digital notch filters performance at different normalized cut-off frequencies allows ensuring that most biomedical signals fit in the tested range, and so the conclusions are applicable to a broad variety of digital biomedical signal processing systems. Although IIR filters are known for their phase distortion, the effect of notch filters with a very high quality factor is very limited in frequency and concentrated only in the stop band [22–26], so it is no major concern. Furthermore, techniques for compensation of distortion have been published [27]. Thus the approach taken is valid.

Subsequently, MATLAB processing capabilities are used to evaluate the fixed-point arithmetic numerical accuracy requirements to realize several types of IIR notch filters, at different design specifications and structures. The performance of a Xilinx FPGA of the Virtex 5 family, when subject to these structural modifications, was assessed using Xilinx ISE 10.1. The discussion and conclusions on the FPGA behaviour complete this work.

2. Second order filters

Using dedicated filter design software, floating-point double precision coefficients were computed for the following filter types: Butterworth, Chebyshev types I and II, and Elliptic. A normalized notch frequency vector $\boldsymbol{\Omega}_{test}$ was considered with 9 points per decade spaced from 10^{-4} to 0.3 (totalling 30 points) and a quality factor vector $\boldsymbol{Q}_{test}$ also with 9 points per decade spaced from 1 to 10^4 (totalling 37 points) and filters of even orders from second to tenth were designed.

In view of the fact that the quantization induces pole movement, a stable filter after quantization may become unstable or even if the quantized filter is confirmed to be stable, its outcome may be unacceptable, thus stating that although the poles remain in the interior of the unit circle, the quantization is too coarse and the poles and zeros movement deforms the filter behavior.

To diminish the wandering of poles and zeros, one valuable method is the implementation of the filter in second-order sections (decomposing an N^th^ order filter in the product of N/2 second order filters, provided that N is even), considering that the coefficients' quantization causes minor pole movement than in higher order sections. The impact of this option will also be evaluated.

2.1. Filter definitions

The normalized frequency Ω is defined as the ratio between the frequency and the Nyquist rate, thus resulting in units of half-cycles per sample.

The quality factor Q is the ratio between Ω_0 and the bandwidth (difference between upper and lower cut-off frequencies Ω_1 and Ω_2), while the notch frequency Ω_0, the centre of the stop band, is the geometric mean of Ω_1 and Ω_2. Since results should be parameterized as functions of Ω_0 and Q and filter design algorithms process Ω_1 and Ω_2, (1) was used to obtain Ω_1 and Ω_2 from design specifications in Ω_0 and Q.

$$\left\{\begin{matrix} \Omega_0 = \sqrt{\Omega_1 \Omega_2} \\ Q = \dfrac{\Omega_0}{\Omega_2 - \Omega_1} \end{matrix}\right\} \Rightarrow \left\{\begin{matrix} \Omega_2 = \dfrac{\Omega_0}{2Q}\left(1 + \sqrt{1+4Q^2}\right) \\ \Omega_1 = \dfrac{{\Omega_0}^2}{\Omega_2} \end{matrix}\right\}. \tag{1}$$

The filters were implemented using both Direct-Form I and II, depicted in Fig. 1, for a second-order section. The respective transfer function, $H(z)$, is presented in (2).

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}. \tag{2}$$

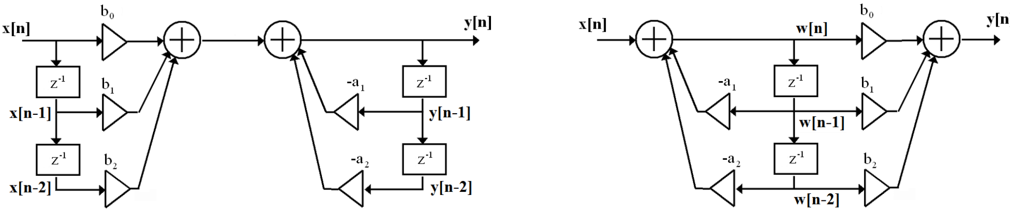

Fig. 1. Implementation of (2): left with Direct-Form I (4), and right with Direct-Form II (5).

Stability assessment was made searching for poles of the filter's transfer function, $H(z)$, outside the unit circle. The n bit fixed-point filter deviations to the floating point double precision format design (16 decimal digits of precision in calculations, IEEE decimal64 format) [28] was measured making use of its frequency response magnitude, $|H_{n\ bit}(j\Omega)|$, root mean square error $\varepsilon_{n\ bit}$ (in dB), using (3),

$$\varepsilon_{nbit} = \sqrt{\sum_{\Omega_{freq\_resp}=\Omega_{\min}}^{\Omega_{\max}} \left[\left| H_{nbit}\left(j\Omega\right) \right|_{dB} - \left| H_{float}\left(j\Omega\right) \right|_{dB} \right]^2}, \tag{3}$$

where $H_{n\_bit}(j\Omega)$ and $H_{float}(j\Omega)$ the transfer functions of both filters, and Ω_{freq_resp} is a vector with 221 points, varying from the 31.8µ half-cycles per sample of $\Omega_{\min}$ up to 1 half-cycle per sample of $\Omega_{\max}$, in which the transfer function discrepancies will be evaluated.

It should be noticed that $\varepsilon_{n\ bit}$ could have been defined in linear units or using the phase or group delay difference, but since the magnitude in dB is the most widely employed method to assess filter response, the parameter $\varepsilon_{n\ bit}$ was chosen to measure directly this dissimilarity in dB. Filter deviations are problematic both in pass and in stop band, since deviations start to manifest in the stop band and afterwards spread to the pass band also. The root mean square (rms) error, defined in (3), copes with this by equally weighting all frequencies.

2.2. Filter structures details

Direct-Form 2 implementation of a digital filter is also known as the canonical form [22–26], this filter realization method uses the minimal number of delay elements, which is equal to the order of the transfer function denominator. The second-order case was presented in Fig. 1 and the transfer function in (2). The difference equation implementation instead of the usual form, which is the base of Direct-Form I, (4), is changed to (5).

$$y[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] - a_1 y[n-1] - a_2 y[n-2], \tag{4}$$

$$\left\{ \begin{array}{l} y[n] = b_0 w[n] + b_1 w[n-1] + b_2 w[n-2] \\ w[n] = x[n] - a_1 w[n-1] + a_2 w[n-2] \end{array} \right\}. \tag{5}$$

Implementing (5) requires 2 delays ($w[n-1]$ and $w[n-2]$) instead the 4 delays ($x[n-1]$, and $x[n-2]$, $y[n-1]$, and $y[n-2]$) required by (4). The number of multiplications is 5, and the number of sums is 4, for both implementations. The difference resides in the amount of computations that may be done in parallel. Using (5) $y[n]$ is computed after $w[n]$, whereas using (4) $y[n]$ is directly computed.

All the delayed signals may be multiplied by their respective coefficient (a_1, a_2, b_1, b_2) in parallel. So, the time sequence of the implementation consists of 2 multiplications (the one stated previously and the product $b_0 w[n]$) and 2 additions (add up $w[n]$ and final computation of $y[n]$).

For Direct-Form I (DF I), as the delays duplicate, the time sequence of the implementation is reduced to half. Therefore, on a parallel computation platform, such as an FPGA, the result is computed in half the time of Direct-Form II (DF II). Another very appealing property of DF I over DF II is that it cannot overflow internally, when fixed-point arithmetic with two’s complement is used, and the output signal is in range [23].

However, for higher order direct-form filters, quantization errors in the filter coefficients grow, especially in notch filters, where the poles and zeros are very close [24], and DF I hardly can be expanded, which is confirmed by the results of Section 4. One further remark is that when quantizing a floating-point filter implemented in second-order Sections, DF I and DF II will have exactly the same behaviour, as both will have the same coefficients. Therefore, the choice of one over the other will depend only on the characteristics of the implementation platform, namely parallel processing capabilities and memory.

3. Second order filter results and discussion

3.1. Filter stability

Second order band-stop filters of the Butterworth, Chebyshev I and II, and elliptic types were implemented using fixed-point arithmetic and the defined $\boldsymbol{Q}_{test}$ and $\boldsymbol{\Omega}_{test}$ vectors. It was found that, for every filter type, only the 16 bit implementation was stable for all ($\boldsymbol{Q}_{test}$, $\boldsymbol{\Omega}_{test}$) pairs. The minimum quality factor to design an unstable filter, Q^{u_min} is 40, in the normalized notch frequency, $\Omega_0^{u\_min}$ was found as 8×10^{-3}.

If the number of bits of the implementation changes the filter stability may vary, in some specific zones of the ($\boldsymbol{Q}_{test}$,$\boldsymbol{\Omega}_{test}$) grid. Generally, designing a different filter type, for the same specification of quality factor, normalized cut-off frequency, and number of bits, will not modify the stability.

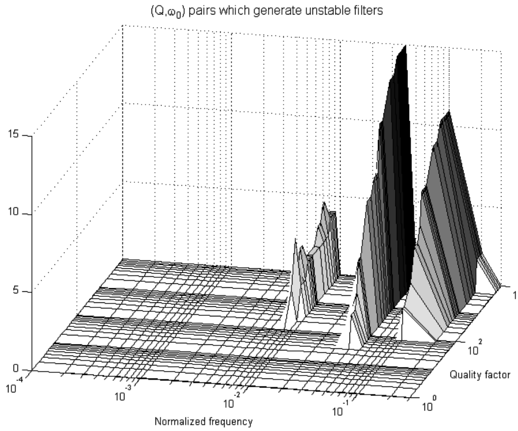

Fig. 2. Number of 2nd order fixed-point unstable filters for each point in the grid ($\boldsymbol{Q}_{test}$,$\boldsymbol{\Omega}_{test}$), in a total of 28 designed per point.

Fig. 2 shows the number of unstable filters obtained for each point in the ($\boldsymbol{Q}_{test}$,$\boldsymbol{\Omega}_{test}$) grid (total of 1110 filters), considering 10 to 16-bit fixed-point implementations of the four filter types (28 designed per point).

Regarding power line notch filter implementation in biomedical systems, the range of the normalized notch frequencies where the filter is unstable represents an important drawback because implementations with sampling rates from 2 kHz down to 200 Hz will cross the two main instability peaks found in Fig. 2. Despite this, if quality factors below 40 are tolerable, the implementation of 10 to 16-bit fixed-point IIR notch filters is straightforward.

3.2. Filter deviations

Second order band-stop filters of the stated types were implemented using the defined $\boldsymbol{\Omega}_{test}$ and $\boldsymbol{Q}_{test}$ vectors. The results obtained for $\varepsilon_{n\ bit}$ in a second order fixed-point Butterworth filter, at a fixed Ω_0 of 0.05, thus situated in the more disturbing zone, are presented in Fig. 3. A grid is displayed with *n* from 10 to 16 bits, and the vector $\boldsymbol{Q}_{test}$.

Smaller values of the quality factor, namely 1, have the higher differences, which is due to the fact that the floating-point filter fixed-point implementation creates a deeper notch than the small deviation due to the fixed-point conversion is able to mimic truthfully. If the quality factor is above 400, the filters have very small notch frequency attenuation and a small amplitude resonance peak both in fixed and floating-point implementations. For quality factor values above 1000 this peak vanishes and the filter acts as an all-pass filter, having no discrepancy from fixed to floating point. To exemplify this behavior in Fig. 4 the magnitude of the 15 bit fixed-point Butterworth filter frequency response, $|H_{15\ bit}(j\Omega)|_{dB}$, is plotted for the $(\boldsymbol{Q}_{test},\boldsymbol{\Omega}_{freq\_resp})$ grid.

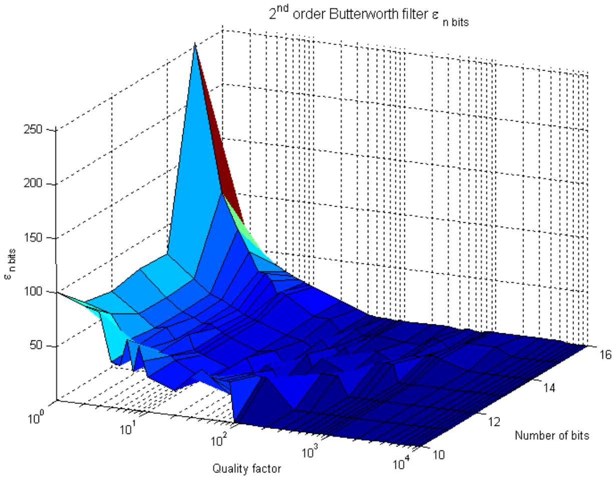

Fig. 3. Second order fixed-point Butterworth filter $\varepsilon_{n\ bit}$. Normalized cut-off frequency of 0.05, n from 10 to 16 bits, and $\boldsymbol{Q}_{test}$ vector.

Butterworth filters are presented as examples in the last two figures, but the other fixed-point filters have exactly the same characteristics regarding the error and the magnitude response progress with the quality factor and all others also generate resonance peaks at very high quality factors.

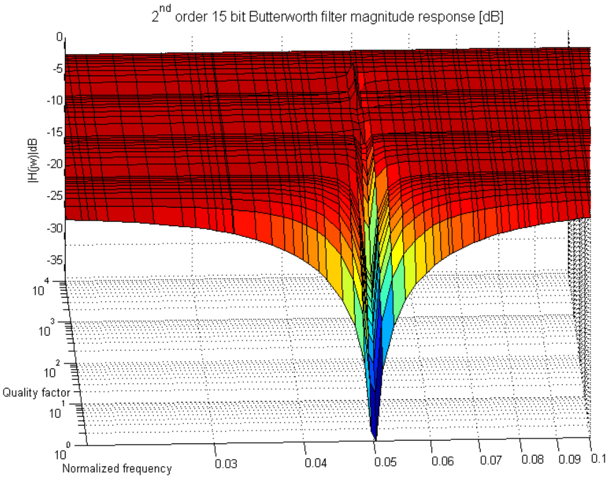

Fig. 4. Magnitude, in dB, of the 15 bit 2nd order fixed-point Butterworth filter, $|H_{15\ bit}(j\Omega)|_{dB}$, with normalized cut-off frequency of 0.05, and using the grid $(\boldsymbol{Q}_{test},\boldsymbol{\Omega}_{freq\_resp})$.

3.3. Filter optimization

In these implementations we searched for the minimum coefficient word length that guaranteed stability and the optimal word length, considering the rms error defined in (3). The filter demanding wider coefficient word lengths to guarantee stability was the elliptic filter. The Chebyshev type I was the most demanding to minimize the mean square error to the floating point implementation. The coefficient word length dependency on quality factor and normalized cut-off frequency for these two cases is shown in Fig. 5.

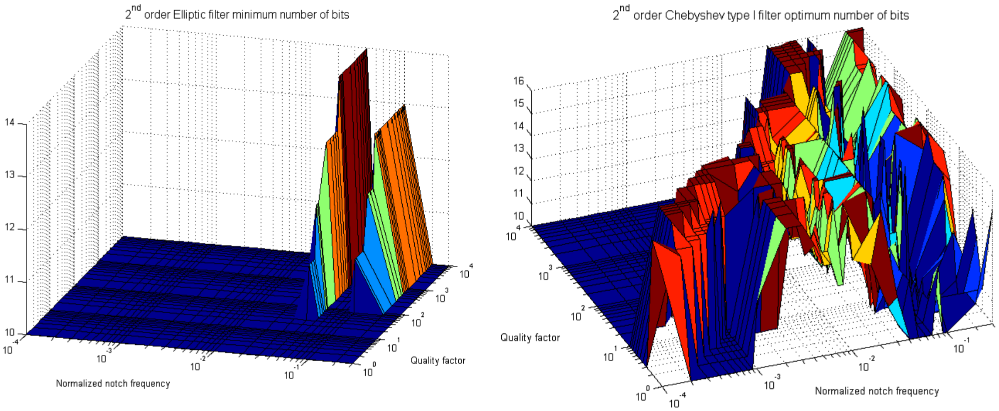

Fig. 5. Minimum coefficient word length to implement a stable 2nd order elliptic filter, for the ($\boldsymbol{Q}_{test}$,$\boldsymbol{\Omega}_{test}$) grid (left). Optimal coefficient word length to implement a stable 2nd order Chebyshev type I filter, minimizing (3), the rms error, for the ($\boldsymbol{Q}_{test}$,$\boldsymbol{\Omega}_{test}$) grid (right).

4. Higher order filter results and dicussion

Repeating the design procedure, 4th, 6th, 8th and 10th order filters were implemented in single section and second-order sections. The results regarding filter stability and quantization effects are subsequently presented.

4.1. Filter stability

Regarding the filter stability, Table 1 presents the number of stable filters designed for each order and each filter type, when using single section (SS) and second-order sections (SOS) implementations of both Direct Forms. The total number of pairs in the ($\boldsymbol{Q}_{test}$,$\boldsymbol{\Omega}_{test}$) grid is 1110, thus 1110 is the maximum number of stable filters possible. The maximum coefficient word length allowed was 16 bits.

Table 1. Number of stable filters of 4th, 6th, 8th and 10th order.

Order	**Type**							
	SOS (DF I and DF II)				**SOS** (DF I and DF II)			
	B	**C1**	**C2**	**E**	**B**	**C1**	**C2**	**E**
4	1110	1110	1110	1110	134	138	162	133
6	1110	1110	1110	1110	16	19	21	16
8	1110	1110	1110	1110	6	7	6	4
10	1110	1110	1110	1105	2	3	2	1

When decomposing the filter structure into second-order sections implementation, the rearrangement of the coefficients allows the minimization of deviations from the poles actual value in such a way that only five elliptic filters of 10th order are unstable, while the remaining 1105 elliptic filters of 10th order are stable. All the other filters of every type and order form 4th to 10th are stable.

In Table 1it is visible that for the 4th order, the single section implementation is no longer valid, since only 12 to 14.6 % of the filters implemented using this structure are stable. For even higher orders even fewer designed filter implementations are characterized by stability.

Other important result is the maximum and the average number of bits required to ensure that the SOS filters are stable for all the pairs in the $(\boldsymbol{Q}_{test},\boldsymbol{\Omega}_{test})$ grid. The results are presented in Table 2.

Table 2. Number of bits (average, maximum, most frequent) to design stable SOS filters for the $(\boldsymbol{Q}_{test},\boldsymbol{\Omega}_{test})$ grid.

Order	SOS filters number of bits to stability											
	Most frequent for all (Q_{test},Ω_{test})				Average for all (Q_{test},Ω_{test})				Median for all (Q_{test},Ω_{test})			
	B	C1	C2	E	B	C1	C2	E	B	C1	C2	E
4	10	10	10	10	10.11	10.13	10.10	10.12	14	14	14	14
6	10	10	10	10	10.13	10.16	10.12	10.17	14	14	14	14
8	10	10	10	10	10.14	10.17	10.15	10.20	14	15	14	16
10	10	10	10	10	10.15	10.18	10.16	10.22	14	15	14	>16

The second-order section filters preserve the behavior presented in Fig. 2. Only a few tens of them require more than 10 bits. When increasing the order the requirements of this residual minority also increase, but only 10 bits are needed for almost every filter implementation.

4.2. SOS Filter deviations

The results of previous Section 2.1 indicate the importance of analyzing not only the global (Q,Ω_0) mesh but also the zones with more demanding coefficient word length to ensure stability. Table 3 and Table 4 summarize some of the measurements made. The first shows the great increase in the number of bits to ensure stability, for a normalized cut-off frequency value of 0.05, thus in the most critical zone. The second represents the average, for n from 10 to 16 bits, of the root mean square error to the floating point implementation, $\varepsilon_{n\ bit}$, defined in (3), for a normalized cut-off frequency of 0.05.

Table 3. Number of bits (average, median, most frequent) to design stable SOS filters for the points in the grid $(\boldsymbol{Q}_{test},\boldsymbol{\Omega}_{test})$ with normalized cut-off frequency of 0.05.

Order	Most frequent for all $(Q_{test},\Omega_{test}=0.05)$				Average for all $(Q_{test},\Omega_{test} = 0.05)$				Median for all $(Q_{test},\Omega_{test} = 0.05)$			
	B	C1	C2	E	B	C1	C2	E	B	C1	C2	E
4	16	16	10	16	13.89	14.54	13.08	13.97	15	15	14	15
6	15	14	10	10	13.57	12.84	12.76	13.16	14	13	13	14
8	10	10	10	14	12.38	12.86	12.41	13.16	12	13	13	14
10	16	16	10	10	13.35	13.59	12.76	12.57	13	14	13	13

The average, median and most frequent coefficients' word length were taken as estimates of central tendency. These three quantities highlight the changes verified on this region, by comparison with Table 2. The averages are significantly larger than generally necessary. The most frequent number of bits increased in several cases, and fluctuations in the median are also noticeable, underlining the volatility of the designs in this region.

Table 4. Average root mean square error, in dB, from the fixed to the floating-point implementation, for a normalized cut-off frequency of 0.05.

Order	**$\varepsilon^{av}(Q_{test},\Omega_{test} = 0.05)$ [dB]**			
	ε_B^{av}	**ε_{C1}^{av}**	**ε_{C2}^{av}**	**ε_E^{av}**
4	90	303	39	254
6	111	193	101	189
8	2502	2500	1105	1124
10	1829	1782	905	1062

The results obtained for the average root mean square error at a fixed normalized cut-off frequency of 0.05, thus situated in the most troubling zone, have their minimum in the Chebyshev type II filter, which has minimum deviations in every order. Chebyshev type II deviations to the floating-point implementation are presented in Fig. 6.

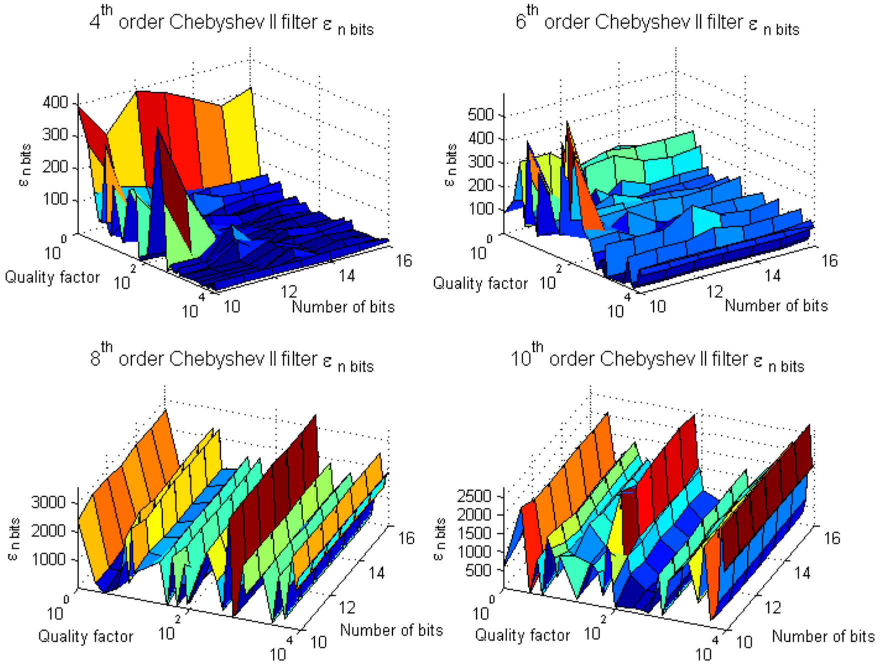

Fig. 6. Higher order Butterworth SOS filters $\varepsilon_{n\ bit}$. Normalized cut-off frequency of 0.05, *n* from 10 to 16 bits, and $\boldsymbol{Q}_{test}$ vector.

4.3. SOS Filter optimization

Table 5 summarizes the coefficient word length, when optimizing this quantity, for the $(\boldsymbol{Q}_{test},\boldsymbol{\Omega}_{test})$ grid, to ensure the minimum root mean square error from the floating-point implementation. It is displayed the coefficients' average, median, and most frequent word length.

Table 5. Number of bits (average, maximum, most frequent) to minimize deviations for the ($\boldsymbol{Q}_{test}$,$\boldsymbol{\Omega}_{test}$) grid.

Order	**Most frequent for all (Q,Ω_0)**				**Average for all (Q,Ω_0)**				**Median for all (Q,Ω_0)**			
	B	**C1**	**C2**	**E**	**B**	**C1**	**C2**	**E**	**B**	**C1**	**C2**	**E**
4	10	10	10	10	12.53	12.73	12.86	12.79	10	12	10	10
6	10	10	10	10	11.84	12.05	11.86	12.15	11	12	10	11
8	10	10	10	10	12.03	12.08	11.91	12.02	10	13	10	11
10	10	10	10	10	12.18	12.24	12.15	12.35	11	12	10	11

The average, median and most frequent coefficients' word length, the three estimates of central tendency taken, illustrate the global behaviour. Although the most frequent is always 10, the fluctuations in the median and in the average, show that this is not the optimum value, and that changes occur among filter types.

Contrary to what one might *a priori* expect, it is seen that the 4th order has the higher average. Behavior verified also for ensuring stability in the critical zone, as previous Table 3 presents. The most demanding filter to minimize the error for all the points in the grid ($\boldsymbol{Q}_{test}$,$\boldsymbol{\Omega}_{test}$) is the 4th order Chebyshev type II filter. The coefficient word length dependence on quality factor and normalized cut-off frequency in these cases is shown in Fig. 7.

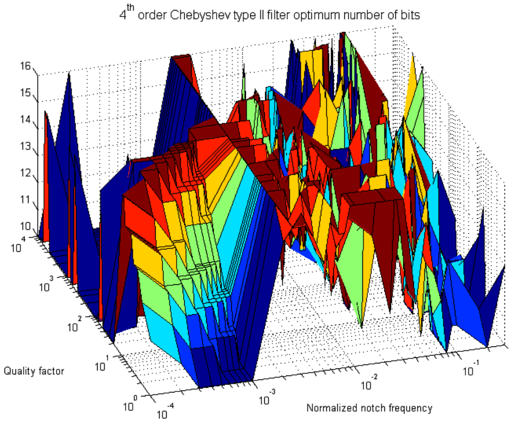

Fig. 7. Optimal coefficients' word length to implement a stable 2nd order Chebyshev type I filter minimizing the norm (3), for the ($\boldsymbol{Q}_{test}$,$\boldsymbol{\Omega}_{test}$) grid.

According to these results, it is not possible to obtain an exact expression to determine the optimal number of bits, from the normalized cut-off frequency and quality factor specifications. This was already suggested by the results of second order, as previous Fig. 5 exemplifies.

5. FPGA Performance

The time sequence of Direct-Form II implementation delay consists of 2 multiplications and 2 additions, whereas in Direct-Form I, the delay necessary to implement a second-order section consists of merely 1 multiplication and 1 addition. Regarding higher order filters implementation, given the previous results, only the performance of SOS realization will be estimated, which will generate an increment in the number of operations proportional to the filter's order increase, *e.g.* 4th order SOS will require 4 multiplications and 4 additions in DF II, and half in DF I.

The additional path delay necessary to link the second-order sections will be disregarded, since it is not relevant, thus top performance will be estimated, hence also the latency will be proportional to the order increase from the 2nd order filter.

5.1. Operation latency

The input data will be considered to have 12 and 16 bits, which is a familiar value in FPGA applications for digital filtering [29], and common in data acquisition boards and analogue to digital converters (ADCs). The results presented were obtained using Xilinx Virtex 5 SX95T speed -3, with timing performance as optimization goal. Pipeline stages were not employed. The fixed point architectures were signed with maximum output precision.

The multiplications have an increase in the latency that would be perfectly linear if the operands always had an even number of bits. However, as seen in Fig. 8, some combinations using one or two odd operands, sometimes have a nonlinear behaviour, differing from the expected.

The additions have even simpler logic, presenting a linear variation in the number of look-up tables with the dimension of the operands, from 1.552 to 1.842 ns. The latency of an addition is less than one third of the respective multiplication.

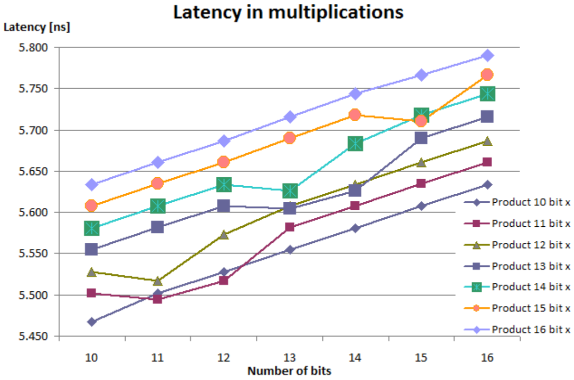

Fig. 8. Multiplication latency for several combinations of operands number of bits with maximum output precision.

All these latencies are very small (below 6 ns), due to the simplicity of the logic, and have a behavior that may be approximated by linear functions with reasonable errors, especially if both the operands have an even number of bits. An even number of bits is standard in commercial ADCs, and from the previous results most filters are optimized with an even number of bits, hence the errors are in fact small.

5.2. Filter frequency of operation

Since the previous results and the considerations taken so far allow the linear interpolation of the delays with minor errors, it is possible to compute the ceiling performances regarding all the combinations of coefficient word length and ADC number of bits.

The filter maximum frequency of operation, f_{MAX}, estimates are presented in the following Table 6 and Table 7, for a 12 bit ADC, based on the FPGA latencies determined in the previous section, and the implementation architecture of each Direct-Form after the place-

and-route process. Table 6 presents the frequency of operation of the filters, regarding average and maximum coefficient number of bits to ensure stability in all points in the ($\boldsymbol{Q_{test}}$,$\boldsymbol{\Omega_{test}}$) grid.

Table 7 displays the frequency of operation associated with the optimum behavior of the filters. The results in terms of coefficient number of bits were presented in Table 2 and Table 5 respectively.

Table 6. Estimated maximum frequency of operation of the filters with coefficient number of bits to ensure stable design for all points in the ($\boldsymbol{Q_{test}}$,$\boldsymbol{\Omega_{test}}$) grid, with 12 bit ADC, in DF I and DF II.

Order	f_{MAX} associated to coefficient number of bits so that all points of the (Q_{test},Ω_{test}) grid are stable [MHz]							
	For mean number of bits in DF I				For mean number of bits in DF II			
	B	C1	C2	E	B	C1	C2	E
2	136.76	136.75	136.77	136.75	71.15	71.14	71.17	71.14
4	68.38	68.38	68.39	68.38	35.58	35.57	35.58	35.57
6	45.58	45.58	45.58	45.58	23.71	23.71	23.71	23.70
8	34.19	34.18	34.19	34.18	17.78	17.78	17.78	17.77
10	27.35	27.35	27.35	27.34	14.22	14.22	14.22	14.21
	For maximum number of bits in DF I				For maximum number of bits in DF II			
2	132.89	132.89	132.89	132.89	67.31	67.31	67.31	67.31
4	132.89	132.89	132.89	132.89	33.65	33.65	33.65	33.65
6	132.89	132.89	132.89	132.89	22.44	22.44	22.44	22.44
8	132.89	132.42	132.89	131.03	16.83	16.66	16.83	16.38
10	132.89	132.42	132.89	---	13.46	13.33	13.46	---

Table 7. Estimated maximum frequency of operation of the filters with coefficient number of bits to minimize deviations for all points in the ($\boldsymbol{Q_{test}}$,$\boldsymbol{\Omega_{test}}$) grid, with 12 bit ADC, in DF I and DF II.

Order	f_{MAX} mean for the (Q_{test},Ω_{test}) grid [MHz]							
	Direct-Form I				Direct-Form II			
	B	C1	C2	E	B	C1	C2	E
2	134.60	134.59	134.41	134.55	68.88	68.87	68.63	68.82
4	67.28	67.23	67.19	67.21	34.42	34.34	34.30	34.32
6	45.03	44.95	45.02	44.93	23.15	23.07	23.14	23.04
8	33.71	33.70	33.75	33.71	17.30	17.29	17.34	17.31
10	26.95	26.95	26.96	26.93	13.82	13.81	13.82	13.79

The previous tables report the improved performance of DF I over DF II. Moreover it is attested that the relative performance variation for different types of filter is very small. The impact of order change in the maximum frequency of operation is coherent with the abovementioned latency change.

The succeeding tables, Table 8 and Table 9, present the estimates, for a 16 bit ADC, of the maximum frequency of operation, obtained with the same methodology as the previous results for a 12 bit ADC. The maximum frequency of operation for any other number of bits of the ADC may be computed by extrapolation, since it was observed that the operations' latency increases linearly. It should be recalled that very few products differ from the linear behaviour, these points may be disregarded in the linear regressions without significant error to extrapolate the latency of other products.

The results of Table 8 and Table 9 show that the increase from 12 to 16 on the number of bits of the ADC is irrelevant for DF I, and of small impact on DF II. Hence, the high speed of computation of the Xilinx device is underlined, especially in lower orders.

Table 8. Estimated maximum frequency of operation of the filters with coefficient number of bits to ensure stable design for all points in the ($\boldsymbol{Q_{test}}$,$\boldsymbol{\Omega_{test}}$) grid, with 16 bit ADC, in DF I and DF II.

Order	**f_{MAX} associated to coefficient number of bits so that all points of the (Q_{test},Ω_{test}) grid are stable [MHz]**							
	For mean number of bits in DF I				**For mean number of bits in DF II**			
	B	**C1**	**C2**	**E**	**B**	**C1**	**C2**	**E**
2	136.76	136.75	136.77	136.75	66.85	66.85	66.86	66.85
4	68.38	68.38	68.39	68.38	33.43	33.43	33.43	33.42
6	45.58	45.58	45.58	45.58	22.28	22.28	22.28	22.28
8	34.19	34.18	34.19	34.18	16.71	16.71	16.71	16.71
10	27.35	27.35	27.35	27.34	13.37	13.37	13.37	13.37
	For maximum number of bits in DF I				**For maximum number of bits in DF II**			
2	132.89	132.89	132.89	132.89	65.91	65.91	65.91	65.91
4	132.89	132.89	132.89	132.89	32.96	32.96	32.96	32.96
6	132.89	132.89	132.89	132.89	21.97	21.97	21.97	21.97
8	132.89	132.42	132.89	131.03	16.48	16.43	16.48	16.38
10	132.89	132.42	132.89	---	13.18	13.14	13.18	---

Table 9. Estimated maximum frequency of operation of the filters with coefficient number of bits to minimize deviations for all points in the ($\boldsymbol{Q_{test}}$,$\boldsymbol{\Omega_{test}}$) grid, with 16 bit ADC, in DF I and DF II.

Order	**f_{MAX} mean for the (Q_{test},Ω_{test}) grid [MHz]**							
	Direct-Form I				**Direct-Form II**			
	B	**C1**	**C2**	**E**	**B**	**C1**	**C2**	**E**
2	134.60	134.59	134.41	134.55	66.29	66.29	66.21	66.27
4	67.28	67.23	67.19	67.21	33.14	33.11	33.10	33.11
6	45.03	44.95	45.02	44.93	22.15	22.13	22.15	22.12
8	33.71	33.70	33.75	33.71	16.60	16.60	16.61	16.60
10	26.95	26.95	26.96	26.93	13.27	13.27	13.27	13.26

5.3. Filter performance analysis

The previous results show that SOS notch filters of 2nd to 10th order can be implemented with a frequency of operation of tens of MHz, even with extreme conditions of quality factor and normalized cut-off frequency. Furthermore, it was verified that the number of bits of the ADC is of small influence in the performance of the FPGA, which also happens with the choice of the type of filter. The reason to this insensibility is the fact that the Virtex-5 device employs DSP48E modules, which are composed of a 25 by 18 bits multiplier, and an adder, and an accumulator of 48 bits, elements with exceeding capacity to deal with operands with the dimension considered for this application.

Changing from DF II to DF I allows the duplication of the working frequency. Therefore, after confirming that the coefficient number of bits is enough to ensure the required filter response, one should choose the implementation of Direct-Form I. The filter type and ADC number of bits, are not constraints in the performance, and it is sure that its implementation in SOS will have a steady frequency of operation and that the filter will be stable. Stability concerns are minor in view of the fact that only 4 elliptic filters of 10th order are unstable. The filter order ends up being the only variable to influence the FPGA implementation performance.

6. Conclusions

In this work the effect of the design specifications was investigated, namely the quality factor and the normalized cut-off frequency, in the number of significant bits necessary to represent the coefficients of an infinite impulse response notch filter. Since implementing these filters using fixed-point arithmetic has much higher accuracy constraints than the common floating-point implementation, the deformations introduced with these specifications were also assessed. The type of structure to implement the filter was also evaluated, and a comprehensive assessment of performance in an FPGA for different ADCs was presented. Since such assessment has never been done, the paper presents an important contribution for the signal processing field, both in notch filter analysis and FPGA implementation.

The first important result found is that it is forbidden to increase the filter's order above the 2nd if the filter is implemented in a single section. However, the order increase is practically harmless if the filter is decomposed into second-order sections. The simulation results obtained provide comprehensive understanding of the stability requirements. Two critical areas, of quality factor and normalized cut-off frequency values, in which filter stability is compromised for some coefficient word lengths, even for 2nd order, were found. These critical areas are especially problematic for biomedical signal processing, since the problematic values of normalized cut-off frequency are typical of these applications, but it will only be important if the quality factor considered is very high, hence not affecting standard applications where a lower quality factor may be tolerated, or even necessary because of slight frequency swings.

The filter deviations were measured and were found to be much increased when going from 6th to 8th and 10th order. From the classical families of IIR filters it was seen that Chebyshev type II is the filter family which suffers less with fixed point implementation, and it is also the less demanding in terms of the average number of significant bits necessary to represent the coefficients. The filter deviations in the critical zone were measured, and were found to increase significantly when rising above the 6th order.

Regarding the minimization of deviations to the floating-point implementation, the average word length is near 12 bits for every filter order, but the optimal number of bits is very dependent on the actual values of quality factor and normalized cut-off frequency. The optimal number of bits, regarding root mean square error minimization, has been seen to define extremely irregular surfaces.

The FPGA implementation of these filters was estimated to be very fast, below 80 ns in a Virtex 5 SX95T-3. The device's performance is insensitive to the number of bits of the ADC, the number of bits of the coefficients used in the filter fixed-point representation, and IIR filter type (Butterworth, Chebyshev, or elliptic). Therefore the filter order and the structure chosen for the implementation (Direct Form I in second-order sections preferably) is almost the unique influent parameter to define filter performance. Maximum frequency of operation is obtained for the DF I structure, as the parallel computation capability of the FPGA extracts the maximum profit from the canonical number of operations performed to compute the output.

It is possible to implement dynamical reconfigurable filters changing the filter specifications of type, coefficients' number of bits, quality factor and normalized cut-off frequency, without disturbing the maximum operating frequency of the FPGA. Even changes on the evolving hardware, for instance input ADC and output DAC, are possible without significant performance variations. This is a very powerful study outcome, since it reveals the possibility of creating a single-chip dynamically reconfigurable digital filter with variable precision, and auto-adaptation properties to minimize numerical errors due to the fixed-point implementation, without significant changes in performance if the order is unchanged.

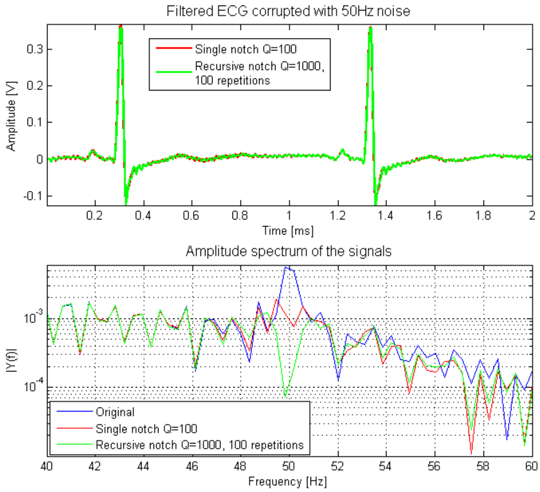

Fig. 9. Example of filtered ECG signal acquired with 50 Hz noise (top), and its amplitude spectrum (bottom). A 4th order Butterworth notch filter with Q =100 was implemented (red), as well as a recursive 2nd order Butterworth notch filter with Q =1000 and 100 repetitions. Real quality factors are very similar, but the recursive presents an attenuation of 37.18 dB while the other only attenuates 17.38 dB.

Moreover, given that, for instance any 2nd order filter in a Xilinx Virtex 5 SX95T-3 requires less than 20 ns to filter the input signal, if the implementation requires a sampling frequency below 1 MHz, the filter could recursively filter its own output at least 50 times, thus completely eradicating the notch frequency from the signal. Such values are perfectly acceptable in several signal acquisition tasks, namely those in biomedical engineering, where notch filters are often necessary, and will not introduce a noteworthy delay. Results for an example ECG signal acquired with a 12 bit ADC are presented in Fig. 9.

Despite having to introduce a number of practical considerations and adjusts in this conceptualized scenario, this study sustains a number of potential new developments using FPGA in the digital signal processing area, and confirms FPGA as a very powerful solution in the analogue signal processing field as well.

Acknowledgements

This work was supported by Instituto de Telecomunicações (IT) and Fundação para a Ciência e Tecnologia (FCT). Eduardo Pinheiro is sponsored by grant SFRH/BD/46772/2008 of FCT.

References

[1] H. Cheng, G. Chiu: “Finite precision controller implementation – limitation on sample rate”. *IEEE/ASME International Conference on AIM*, Kobe, Japan, 2003, pp. 634–639.

[2] V. Davídek, M Antosová, B. Psenicka: “Finite word-length effects in digital state-space filters”. *Radioengineering*, vol. 8, no. 4, Dec. 1999, pp. 7–10.

[3] B. Liu: “Effect of finite word length on the accuracy of digital filters – a review”. *IEEE Transactions on Circuit Theory*, vol. CT-18, no. 6, Nov. 1971, pp. 670–677,

[4] H. Butterweck, J. Ritzerfeld, M. Werter: “Finite wordlength effects in digital filters: a review”. *Fac. Of Elec. Eng*., Eindhoven University of Technology, Netherlands, EUT Report 88-E-205, 1988.

[5] T. Laakso, J. Ranta, S. Ovaska: "Design and implementation of efficient iir notch filters with quantization error feedback". *IEEE Transactions on Instrumentation and Measurement*, vol. 43, no. 3, Jun. 1994, pp. 449–456.

[6] R. Otnes, L. McNamee: "Instability thresholds in digital filters due to coefficient rounding". *IEEE Transactions on Audio and Electroacoustics*, vol. AU-18, no. 4, Dec. 1970, pp. 456–463.

[7] J. Datorro: "The implementation of recursive digital filters for high-fidelity audio". *Journal Audio Engineering Society*, vol. 36, Nov. 1988, pp. 851–878.

[8] M. Er: "Designing notch filters with controlled null width". *Signal Processing*, vol. 24, Sep. 1991, pp. 319–329.

[9] T. Laakso, I. Hartimo: "Noise reduction in recursive digital filters using higher-order error feedback". *IEEE Transactions on Signal Processing*, vol. 40, May 1992, pp. 1096–1107.

[10] J. Wilkinson: *Rounding Errors in Algebraic Processes*. Englewood Cliffs, New Jersey, 1963.

[11] G. Forsythe, C. Moller: *Computer Solution of Linear Algebraic Systems*. Englewood Cliffs, New Jersey, 1967.

[12] B. Widrow: "Statistical analysis of amplitude quantized sampled-data systems". *AIEE Transactions on Applications and Industry*, vol. 79, Jan. 1961, pp. 555–568.

[13] J. Knatzenelson: "A note on errors introduced by combined sampling and quantization". *IRE Transactions on Automatic Control*, vol. AC-7, Apr. 1962, pp. 58–68.

[14] J. Knowles, R. Edwards: "Effects of a finite-word-length computer in a sampled-data feedback system". *IEE Proceedings.*, vol. 112, Jun. 1965, pp. 1197–1207.

[15] C. Weaver, J. Van der Groeben, P. Mantey, J. Toole, C. Cole Jr., J. Fitzgerald, R. Lawrence: "Digital filtering with applications to electrocardiogram processing". *IEEE Transactions on Audio and Acoustics*, vol. AU-16, no. 3, Sep. 1968, pp. 350–391.

[16] K. Liu, R. Skelton, K. Grigoriadis: "Optimal controllers for finite wordlength implementation". *IEEE Transactions on Automatic Control*, vol. 37, no. 9, Sep. 1992, pp. 1294–1304.

[17] G. Yan: "New digital notch filter structures with low coefficient sensitivity". *IEEE Transactions on Circuits and Systems*, vol. CAS-31, no. 9, Sep. 1984, pp. 825–828.

[18] R. Goodal: "A practical method for determining coefficient word length in digital filters". *IEEE Transactions on Signal Processing*, vol. 40, no. 4, Apr. 1992, pp. 981–985.

[19] J.F. Kaiser: "Digital filters". *System Analysis by Digital Computers*, F.F. Kuo, J.F. Kaiser, Eds. New York, Wiley, 1966.

[20] P. Mantey: "Eigenvalue sensitivity and state-variable selection". *IEEE Transactions on Automatic Control*, vol. AC-13, no. 3, Jun. 1968, pp. 263–269.

[21] E.C. Pinheiro, O. Postolache, P.M. Girão: "Digital Notch Filters Implementation with Fixed-point Arithmetic". *Proc. XIX IMEKO World Congress*, Lisbon, Portugal, Sept. 2009, pp. 491–496.

[22] A.V. Oppenheim, R.W. Schafer: *Discrete-Time Signal Processing 3rd Edition*. Englewood Cliffs, New Jersey, Prentice-Hall, 2009.

[23] J.O. Smith: *Introduction to Digital Filters with Audio Applications*. W3K Publishing, 2007.

[24] G. Dahlquist, Å. Björck: *Numerical Methods*. Englewood Cliffs, New Jersey, Prentice-Hall, 2003.

[25] A. Antoniou: *Digital Filters – Analysis, Design, and Applications*. New York, USA, McGraw-Hill, 2000.

[26] L.B. Jackson: *Digital Filters and Signal Processing*. New York, USA, Springer, 2010.

[27] M. Kozioł: "Frequency-response compensation of distorting parts of measurement systems". *Metrol. Meas. Syst.*, vol. XIII, no. 3, 2006, pp. 263–276.

[28] IEEE, *IEEE 754-2008 Standard for Floating-Point Arithmetic*, August 2008.

[29] D. Gallo, C. Landi, M. Luiso: "Compensation of current transformers by means of field programmable gate array". *Metrol. Meas. Syst.*, vol. XVI, no. 2, 2009, pp. 279–288.