

Pseudo-random Sequence Based Tuning System for Continuous-time Filters

F. Corsi*, C. Marzocca*, G. Matarrese*, A. Baschirotto[°], S. D'Amico[°]

* Dipartimento di Elettrotecnica ed Elettronica, Politecnico di Bari, Italy

[°] Dipartimento di Ingegneria dell'Innovazione, Università di Lecce, Italy

Abstract

Continuous-Time filters are widely used in signal processing but require a tuning system to align their frequency response. Several tuning techniques have been proposed in the literature, which can be grouped in two basic schemes: master-slave and self-calibration arrangements. Here we propose a novel tuning approach which can be applied to both tuning schemes. The tuning algorithm is based on the application of a pseudo-random input Test Pattern Signal and on the evaluation of a few samples of the input-output cross-correlation function. The key advantages of the proposed technique are basically the use of a pseudo-random pattern signal which can be generated by a very simple circuit in a small die area and the simple circuitry required to sample the filter output and to perform the cross-correlation operation.

Some experimental results of the application of the proposed tuning technique to a benchmark filter are given in order to assess its effectiveness.

1. Introduction

In Continuous-Time (CT) filters the time-constants are defined by uncorrelated components (g_m/C or $R \cdot C$), therefore, even if they can operate with low op-amp UGB (unity-gain-bandwidth), they strongly require a tuning system to align their frequency response. This becomes particularly important when compensating for component variation from their nominal values (due to technological spread, aging, temperature, etc.), and when aligning the filter frequency response to different target frequency responses as required, for instance, by a multistandard telecommunication terminal.

Several tuning techniques have been proposed in the literature to adjust continuous-time filter frequency response [1-6]. They can be described and compared in terms of frequency response accuracy, time for tuning, and interactions with filter performance.

The popular master-slave tuning scheme, is composed by a slave filter which has to effectively process the signal and by a master filter just for tuning purpose which has to

be properly matched with the slave circuit [3]. In the master circuit a time constant is adjusted by controlling one or more component values with a tuning control signal, which is then passed to the matched slave filter, whose frequency response is then properly adjusted in order to process the true input signal. The control signal can be either analog or digital, presenting advantages and disadvantages, as follows.

An analog control signal allows to continuously adjust the time constant with an analog control voltage. This is the case of g_m -C or MOSFET-C filters, in which the 'resistive' part of the time constant (transconductance g_m or MOSFET resistance) can be controlled by an analog voltage/current. This approach presents all the typical advantages of analog systems (large not-quantized accuracy, no distributed clock, etc.). On the other hand, it requires that the tuning system is continuously active (thus increasing the overall power consumption), since storing a high accuracy analog control voltage would be difficult.

In the alternative tuning approach, a digital word is employed to switch on or off a set of analog unit element devices, which determine the time-constant to be tuned [4], and the accuracy of the tuning system is correlated to the quantization of the component whose value has to be adjusted. Thus, the length (number of bit) of the control word typically corresponds to the achievable system accuracy.

The master-slave tuning scheme is effective only if a good matching between the master and the slave circuits is possible. If this requisite is not satisfied, this scheme is no more practical. For this reason a self-calibrating scheme can be adopted [5]. In this solution the transfer function of the main filter is directly controlled. The main filter operates in two separate configurations: calibration or signal processing. During calibration, a Test Pattern signal is applied to its input and the relative output voltage is processed by the tuning circuitry in order to adjust the control signal. This control signal is used during signal processing to maintain the adjusted filter frequency response. This solution is particularly effective with a digital approach, in which the control words are memorized in proper registers. On the other hand, this approach requires a specific timing scheme, since the signal cannot be processed during calibration phase,

which could be a strong limitation in certain signal processing systems. For both tuning schemes (master-slave and self-calibration) the algorithm to be implemented for the time constant control has then to be discussed. In most tuning algorithms, the main parameters, which can be used to compare them, are the kind of input signal pattern and complexity of its generation circuitry, the control of a single parameter of the frequency response (gain, pole frequency, etc.) or of the complete frequency response behavior, the error detection circuit complexity and accuracy, and the control algorithm circuit complexity and accuracy.

In the following, a novel tuning approach is proposed [7]. This scheme can be applied to both the master-slave and to the self-calibration scheme. The tuning algorithm is based on the use of a pseudo-random Test Pattern signal and on the evaluation of a few samples of the cross-correlation between the pseudo-random Test Pattern and the corresponding filter response [7]. The proposed technique offers several advantages. First, the pseudo-random input pattern signal can be generated with a very simple circuit in a small die area. Moreover, the tuning of the overall filter frequency response is carried out (and not of a part matched with it). Last, a very simple tuning circuitry is required for the evaluation of the cross-correlation samples needed by the tuning scheme [7].

2. The pseudorandom sequence based tuning algorithm

For a Linear Time Invariant (LTI) circuit, the tuning parameters are directly related to its impulse response $h(t)$ and the tuning process can be accomplished through the evaluation of a suitable approximation of $h(t)$. In particular, it has been shown [8] that the main circuit specifications can be related to a limited number of samples of $h(t)$.

It has been widely reported in the literature [9] that the impulse response $h(t)$ of an LTI circuit can be estimated by means of the input-output cross-correlation function $R^{xy}(t)$, provided the input signal has the frequency spectrum of a white noise. For continuous-time signals and stationary input $x(t)$, the cross-correlation function $R^{xy}(t)$ is given by:

$$R^{xy}(t) \cong \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T x(\tau)y(t+\tau)d\tau, \quad (1)$$

where $y(t)$ is the circuit response. When the input signal $x(t)$ is ergodic, we can express the cross-correlation function as a statistic average (expected value):

$$R^{xy}(t) = E[x(\tau)y(t+\tau)] = \int_{-\infty}^{+\infty} h(\theta)R^{xy}(t-\theta)d\theta = R^{xx}(t) * h(t), \quad (2)$$

where $R^{xx}(t)$ represents the auto-correlation function of $x(t)$. When the input is an ideal white noise, $R^{xx}(t)$ is a Dirac pulse $\delta(t)$, therefore $R^{xy}(t)$ coincides with $h(t)$.

In a typical on-chip implementation of the cross-correlation algorithm, $x(t)$ is a finite length sequence of L rectangular pulses of constant width Δt and whose amplitude can assume a positive or negative value with same probability.

In the ideal case, the auto-correlation function of such a signal is:

$$R^{xx}(t) = \lim_{L \rightarrow \infty} \frac{1}{2L+1} \sum_{n=-L}^L \frac{1}{\Delta t} \int_{n\Delta t}^{(n+1)\Delta t} x(\tau)x(t+\tau)d\tau. \quad (3)$$

In practice, even when $L \rightarrow \infty$, $R^{xx}(t)$ is only an approximation of the Dirac's pulse, since its shape is a triangular pulse of width depending on Δt :

$$R^{xx}(t) = \begin{cases} 1 - \frac{t}{\Delta t} & |t| < \Delta t \\ 0 & |t| \geq \Delta t \end{cases} \quad (4)$$

Furthermore, the finite length of the sequence introduces a tail in the auto-correlation function of the input stimulus and this affects the accuracy of the estimated $h(t)$, especially for large values of t . As a consequence the width of the single pulse Δt and the length L of the sequence must be carefully chosen, depending on the bandwidth of the circuit and on the accuracy of the estimation of $h(t)$ required by the tuning operation [8]. Moreover, the power density spectrum of $R^{xx}(t)$ as expressed in equation (3) exhibits the first zero at $f_0 = 1/\Delta t$, and thus, for the accurate tuning of a filter with cut-off frequency f_c , f_0 should be chosen conveniently greater than f_c . We usually assume $f_c \cong f_0/5$, so that the width of each pulse of the pseudo-random sequence should be $\Delta t \leq 1/5f_c$.

A Linear Feedback Shift Register (LFSR) with a suitable number of stages can perform the on-chip generation of the pseudo-random sequence. The use of a LFSR entails the advantage that, once the initial states of the flip-flops have been set, the finite input sequence is univocally determined, thus the tail of its auto-correlation function is always the same and the input-output cross-correlation function depends only on the current tuning configuration of the filter. In other words, the tails of the auto-correlation function of the input sequence do not depend on the particular realization of a stationary and ergodic process $x(t)$.

In the typical scheme shown in Fig. 1, used for the on-line tuning, the filter is embedded between a DAC and an ADC. In the following it will be shown that simple assumptions strongly simplify the system implementation, making it very advantageous.

First of all, since the pseudo-random Pattern Signal generator delivers a two-level bitstream, the DAC is realized with two switches which connect the filter input node to one of two reference voltages. It can be shown that the auto-correlation function of the two-level input sequence achieved is a good approximation of a Dirac's delta.

On the other hand, the quantization error of the ADC could affect the evaluation of the input-output cross-

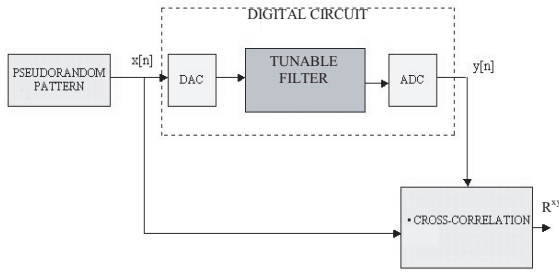


Fig. 1. Evaluation of the cross-correlation function.

correlation function. However, we found that even a very low resolution ADC may be employed as the quantization errors are averaged by the cross-correlation operation. This strongly simplifies the system implementation. Thus, by increasing the length of the input sequence, even a 1-bit ADC, i.e. a simple comparator, can be employed, for tuning purposes. This is of particular relevance when a high sampling rate is needed since, in general, high resolution ADCs cannot be too fast. Fig. 2 shows different estimations of the cross-correlation function of a Butterworth low-pass filter, obtained for three input sequences of different lengths and using a 1-bit ADC to sample the output.

As a conclusion, the system of Fig. 1 can be implemented for the specific case of filter tuning with a single bit DAC and a single bit ADC, which greatly reduces the cross-correlation circuit complexity [7].

Once a good estimation of the filter impulse response has been achieved, by means of the input-output cross-correlation function, the next issue to be dealt with in the tuning operation is the choice of a suitable set of samples of the cross-correlation function assumed as circuit signature.

Starting from the specifications to be tuned, $s_j, j=1, \dots, n$, the most advantageous situation would occur if the tolerance interval defined on each s_j would map monotonically in a corresponding interval of at least one of the n samples of the signature. In this case the procedure would only consist in tuning one specification at a time, on the basis of the comparison between the actual value of the corresponding R^{xy} sample and the one of the nominal circuit [10,11].

If this situation does not occur, a study of the sample sensitivities with respect to the filter specifications s_j is required. These sensitivities may be expressed in terms of the partial derivatives $\partial_{ij} = \partial R^{xy}(m_i) / \partial s_j$. In this way it is possible to identify the conditions to be met in the choice of the signature, in order to approximate the ideal above described mapping [10,11].

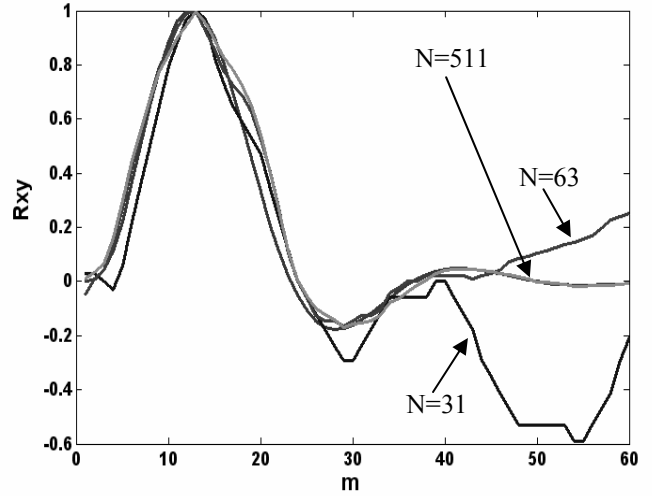


Fig. 2. Butterworth low-pass filter: normalized cross-correlation functions obtained with a 1-bit comparator and input sequences of different lengths.

Let us consider a filter characterized by a performance parameter vector $S=(s_1, s_2, \dots, s_n)$, and suppose that \bar{S} is the vector of the nominal specifications; if a sample of the cross-correlation function $R^{xy}(m_i)$ exists in which the following conditions are met:

$$\partial_{11}(\bar{S}) \gg \partial_{12}(\bar{S}), \dots, \partial_{1n}(\bar{S}), \quad (5)$$

it has been shown [10,11] that, with reference to e.g. the specification s_1 :

$$R^{xy}(S, m_1) > R^{xy}(\bar{S}, m_1) \Rightarrow s_1 > \bar{s}_1 \quad \text{if } \partial_{11} > 0. \quad (6)$$

$$R^{xy}(S, m_1) < R^{xy}(\bar{S}, m_1) \Rightarrow s_1 < \bar{s}_1$$

Thus it is possible to tune the filter specification s_1 to its nominal value, by means of the value of the m_1^{th} sample of the cross-correlation function, as compared to the one of the nominal circuit [7].

Once the specification s_1 has been tuned, the next sample m_2 of the signature can be chosen as the one which satisfies the less restrictive conditions:

$$\partial_{22}(\bar{S}) \gg \partial_{23}(\bar{S}), \dots, \partial_{2n}(\bar{S}).$$

In other words, since the first specification s_1 has already been tuned, the second sample $R^{xy}(S, m_2)$ can be chosen regardless of the value of ∂_{21} .

By iterating this procedure, the complete signature is identified as the minimum set of samples needed to tune

the filter specifications to their nominal values within fixed tolerances [7].

3. Tuning of a benchmark active-RC filter

Usually the time constant deviations, and thus the cut-off frequency of an active-RC filter, can be corrected by using switchable binary-weighted capacitor arrays with MOSFET switches. The capacitor array, whose total capacitance is here indicated by C_{array} , consists of a fixed capacitance C_{off} and N binary weighted switchable elements, the smallest of which has a value denoted δC . The capacitor array can be tuned by a N bit digital code in order to compensate the technological spread. The total capacitance value achieved for a given tune code is:

$$C_{array} = C_{off} + n \cdot \delta C, \quad (7)$$

where n is an integer in the range $[0, (2^N-1)]$. Given a x per cent technological spread, the design equations are:

$$C_{off} = \frac{C_{nom}}{1 + x/100}, \quad (8)$$

$$\delta C = \frac{1}{2^N - 1} \left(\frac{C_{nom}}{1 - x/100} - C_{off} \right). \quad (9)$$

The maximum error between the discrete capacitance value C_{array} and the nominal value C_{nom} is given by:

$$\varepsilon_{MAX} = \pm \frac{1}{2} \frac{\delta C}{C_{nom}} 100\%. \quad (10)$$

Both the fixed capacitance C_{off} and the switchable elements δC are proportional to the nominal value C_{nom} , as it is apparent from equations (8-9). As a consequence, two different capacitor arrays on the same chip, tuned by the same digital code, exhibit the same percentage variation.

In practice the ratio between the two capacitor arrays remains constant also after they have been tuned.

The proposed tuning system [7] has been employed for the alignment of the frequency response of an anti-aliasing filter for base-band UMTS receivers [12]. The filter has a 4th order low-pass Butterworth transfer function with a 2MHz pole frequency and a unit DC-gain. To meet all the requirements, a RC-active filter having a multipath biquadratic cell as building block is employed. The filter is then the cascade of two fully differential multipath biquadratic cells.

Each multipath cell allows to use only one op-amp for each couple of poles with a consequent power saving. The transfer function of the single cell is given by:

$$T(s) = \frac{-G_1 G_2}{C_1 C_2 s^2 + C_2 (G_1 + G_2 + G_3) s + G_2 G_3}, \quad (11)$$

with $R_1=1/G_1$, $R_2=1/G_2$, $R_3=1/G_3$. The characteristic parameters of each cell are: the static gain A_v , the quality factor Q and the pole frequency ω_c and are given by:

$$A_v = \frac{-G_1 G_2}{G_2 G_3}; \quad Q = \sqrt{\frac{C_1}{C_2}} \cdot \frac{1}{\frac{G_1}{\sqrt{G_2 G_3}} + \sqrt{\frac{G_2}{G_3}} + \sqrt{\frac{G_3}{G_2}}}; \quad (12)$$

$$\omega_c = \sqrt{\frac{G_2 G_3}{C_1 C_2}}.$$

Table I gives the nominal values of each component.

Notice that the static gain and the pole quality factor depend only on ratios of homogeneous components, while the pole frequency depends on non-homogeneous components. Changing the value of the capacitors C_i allows to control the value of ω_c , which is thus the only specification to be tuned. This can be done realizing these components as arrays of unit elements whose connection is given by a digital word, as described above. The number of unit elements (i.e. the number of control bits) gives the accuracy of control of the passive components and, therefore, of the parameter (ω_c) to be tuned.

For typical receiver channels, the pole frequency is required to exhibit a relative error with respect to its nominal value lower than 5%. This will depend on the number of bits adopted for the digital tuning.

TABLE I. Components and characteristics of each cell.

Cell components and characteristics	1 st Cell	2 nd Cell
$R_1=R_2=R_3$	10k Ω	10k Ω
C_1	12.24pF	29.57pF
C_2	4.65pF	1.92pF
Pole-frequency	2.11MHz	2.11MHz
Q	0.54	1.3

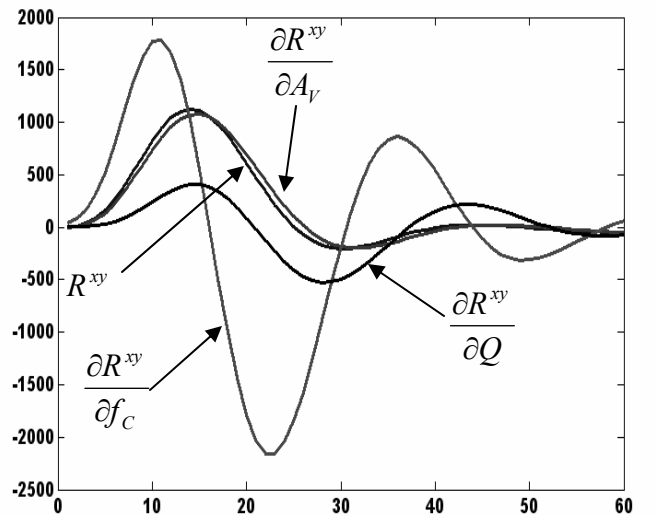


Fig. 3. Diagrams of R^{xy} and of its derivatives for a Butterworth low pass filter.

According to the considerations of sec. 2, only one sample $R^{xy}(m)$ of the cross-correlation function can be used as a signature of the circuit, since there is only one specification to be adjusted. This sample should be the one which satisfies the conditions:

$$\frac{\partial R^{xy}(m)}{\partial f_c} \gg \frac{\partial R^{xy}(m)}{\partial A_v}, \frac{\partial R^{xy}(m)}{\partial Q}. \quad (13)$$

Fig. 3 shows, with reference to a 4th order Butterworth filter, the diagrams of the cross-correlation function R^{xy} and of its partial derivatives with respect to A_v , Q and f_c . It results, from Fig. 3, that $R^{xy}(m)$ should be chosen with m in the range [20-23], as in this range the partial derivative of R^{xy} taken with respect to f_c assumes a much greater value than the other partial derivatives.

However the choice of the first zero-crossing sample of R^{xy} ($m=25$) as signature would greatly simplify the hardware implementation of the tuning algorithm. In fact, in this case the tuning operation would be performed considering only the sign of this sample. Fig. 4 shows the plot of the 25th sample of the cross-correlation function vs the cut-off frequency f_c . The shape of this plot suggests a simple tuning algorithm [7]: if, for instance, $R^{xy}(25)$ assumes a negative value for a given configuration of the tuning capacitor array, this also implies that $f_c > 2.11$ MHz, thus C_{array} must be increased to approach the nominal cut-off frequency. A possible simple procedure consists in adding one δC at a time until the sample $R^{xy}(25)$ becomes positive [7]. Then the best tuning configuration would be the one corresponding to the smaller absolute value of the signature sample. In the worst case, with this approach, the cross-correlation operation must be performed at most 2^{N-1} times, if the tuning operation starts from an initial configuration $C_{array} = C_{off} + (2^{N-1} - 1)\delta C$. A binary search algorithm can be also used to find the best tuning configuration. In this case, the evaluation of the signature

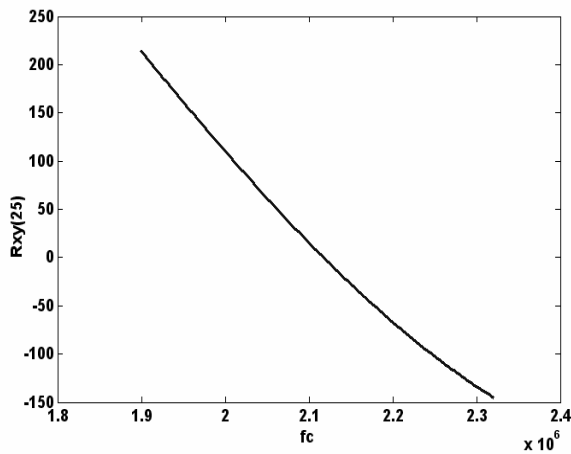


Fig. 4. Plot of the 25th sample of R^{xy} vs the cut-off frequency f_c for the benchmark filter. The remaining specifications are fixed to their nominal values.

sample should be performed at most only N times, therefore the tuning operation can be made faster. However a digital finite state machine must be added to implement the binary search algorithm, therefore the convenience of this solution must be assessed considering also this extra-hardware needed.

4. Simulation results

In order to evaluate the efficiency of the proposed tuning procedure, a set of MATLAB worst case simulations have been carried out on a high-level description of the benchmark filter. A technological spread of $\pm 20\%$ has been considered both for the resistors and for the capacitors. The arrays have been sized according to eq. (7-9), and considering a global technological spread $\alpha = 40\%$. The cross-correlation function has been estimated according to the scheme in fig. 1, using a 1-bit ADC. The length of the input sequence is not critical in the tuning operation, since cross-correlation estimations obtained with LFSR of different number of stages differ practically only in the tails (see fig. 2). Thus a LFSR of 6 stages ($L=63$ and $f_0=5f_c$) has been used in the simulations. In every simulation the filter has been tuned following the first scheme proposed in sec. 3, that is by adding or subtracting one δC at a time to the capacitor arrays.

Fig. 5 shows the cross-correlation functions and the Bode diagrams obtained in correspondence of the two worst case values, f_{cmin} and f_{cmax} of the cut-off frequency; the nominal and the tuned curves are also reported.

Table II summarizes the results of the simulations in terms of the maximum relative errors on the tuned cut-off frequency, with different numbers of tuning bits N . The minimum values of the tuning capacitors δC are also reported. From table II it appears that increasing N over 5 does not cause a further decrease in the maximum error, since the overall accuracy is limited by the accuracy in the estimation of the cross-correlation function.

Table II. Summary of the tuning results for the benchmark filter.

N	Resistor Variation [%]	Capacitor Variation [%]	δC	Maximum Error on f_c [%]
3	15	12	262 fF	8.68
4	12	20	122 fF	4.15
5	20	20	59 fF	2.79
6	20	20	29 fF	2.79

Finally an estimation of the time needed by a single tuning operation should be done. Basically the tuning time is dominated by the time t_{IN} needed to apply the input pseudorandom sequence, which depends on its length L

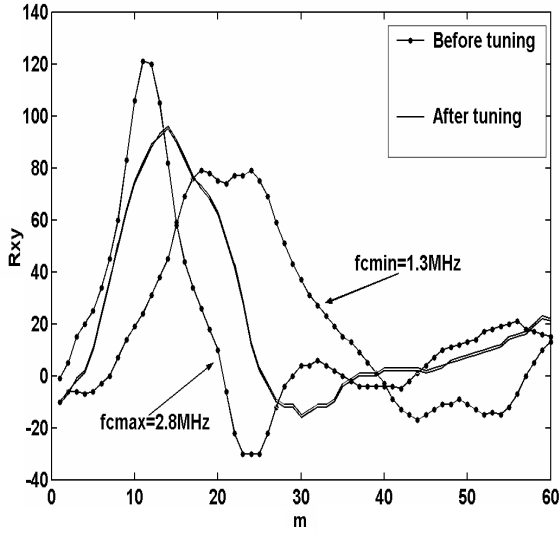


Fig. 5. Cross-correlation functions and Bode diagrams corresponding to the worst-case configurations before the tuning operation.

and on the width of the single pulse Δt . For a six stage LFSR ($L=63$), the time t_{IN} is:

$$t_{IN} = L\Delta t = L \frac{1}{f_0} = L \frac{1}{5f_c} = 63 \frac{1}{5 \times 2.11MHz} \cong 6\mu s. \quad (14)$$

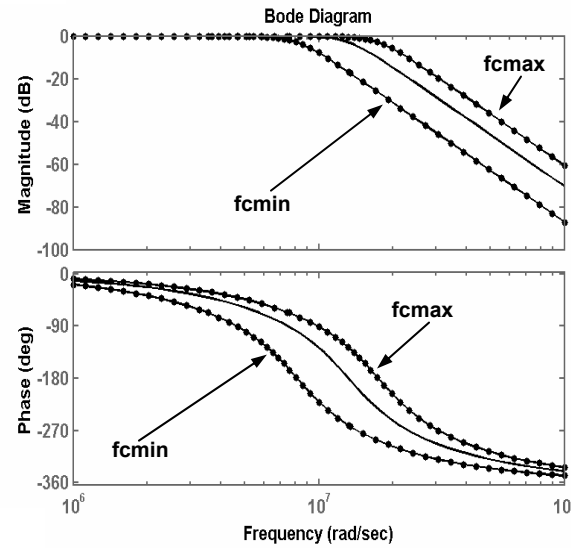
The total tuning time, t_{TOT} , depends on the number of cross-correlation estimations needed by the whole tuning operation. Thus for the two tuning procedures described in sec. 3 we have, respectively $t_{TOT}=(2^{N-1}) \times t_{IN}$, or $t_{TOT}=N \times t_{IN}$.

5. Conclusions

In this paper a novel technique for the tuning of continuous-time filters is proposed. The technique is based on the use of a pseudo-random test pattern to measure, and, as a consequence, correct, the filter frequency response. The key issues of this proposal regard the simplicity of the input test pattern generation and also of the measuring output signal circuitry, which is limited to a single comparator. The output error evaluation is realized with a cross-correlator operating with a few output samples sequence. The proposed technique allows to achieve a tuning error mainly dominated by the number of bit used in the tuning circuit and with a very small tuning time. This appears particularly interesting for the application of the proposed technique to the realization of telecommunication multi-standard terminals. Extension to high-Q, continuous-time filters is currently under study.

References

[1] Y. P. Tsividis, "Integrated Continuous-Time Filter Design- An Overview", *IEEE J. Solid-State Circuits*, Mar. 1994, pp. 166-176.
 [2] T. K. Pham, P. E. Allen, "A Design of a Low-Power, High-Accuracy, and Constant-Q-Tuning Continuous-Time Bandpass



Filter", *IEEE International Symposium on Circuits and Systems, 2002 (ISCAS 2002)*, vol. 4, pp. 639-642, May 2002.

[3] V. Gopinathan, Y.P. Tsividis, K.-S. Tan, and R.K. Hester, "Design Considerations for High-Frequency Continuous-Time Filters and Implementation of an Antialiasing Filter for Digital Video", *IEEE Journal of Solid State Circuits*, Dec. 1990, pp. 1368-1378

[4] A. M. Durham and W. Redhman-Whhe, "Integrated continuous-time balanced filters for 16-b DSP interfaces", *IEEE J. Solid-State Circuits*, vol. 28, pp. 835-839, July 1993.

[5] J.B. Hughes, N.C. Bird, R.S. Soin, "A Novel Digitally Self-Tuned Continuous-Time filter Technique", *IEEE Intern. Symp. On Circuits and Systems, (ISCAS 1986)*, pp. 1177-1180.

[6] T. Sümesağlam, A. İ. Karşilayan, "Digital tuning of high-order high-Q continuous-time filters", *IEE Electronics Letters*, vol. 38, 12 September 2002, pp. 1076-1078.

[7] A. Baschiroto, F. Corsi, S. D'Amico, A. Digiandomenico, C. Marzocca, G. Matarrese, "Pseudo-random Sequence Based Tuning System for Continuous-time Filters", *USA Patent (pending)*, 2003.

[8] C. Marzocca, F. Corsi, "Mixed-Signal Circuit Classification in a Pseudo-Random Testing Scheme", *Journal of Electronic Testing: Theory and Applications (JETTA)*, vol. 18, pp. 333-342, 2002.

[9] C. Y. Pan, K. T. Cheng, "Pseudo-Random Testing for Mixed-Signal Circuits", *IEEE Trans. on Computer-Aided Design*, vol. 16, n. 10, 1997.

[10] F. Corsi, C. Marzocca, G. Matarrese, "An approach to the classification of mixed-signal circuits in a pseudorandom testing scheme", *Proceedings of the IEEE Design, Automation and Test in Europe Conference and Exhibition (DATE '03)*, pp. 1178-1179, 2003.

[11] F. Corsi, C. Marzocca, G. Matarrese, "Defining a bist-oriented signature for mixed-signal devices", *Proceedings of the IEEE Southwest Symposium on Mixed-Signal Design*, pp. 202-207, 2003.

[12] D. Manstretta, R. Castello, F. Gatta, P. Rossi, F. Svelto, "A 0.18 μ m CMOS Direct-conversion receiver front-end for UMTS", *IEEE International Solid-State Circuits Conference*, vol. XLV, pp. 240 - 241, February 2002.