Polyphase FIR networks based on frequency sampling for multirate DSP applications

Fernando Cruz–Roldán, José David Osés del Campo,

Juan Ignacio Godino–Llorente, Luciano Boquete–Vázquez, and C. J. Bleakley,

Abstract

It is well known that the frequency sampling approach to the design of Finite Impulse Response digital filters allows recursive implementations which are computationally efficient when most of the frequency samples are integers, powers of 2 or nulls. The design and implementation of decimation (or interpolation) filters using this approach is studied herein. Firstly, a procedure is described which optimizes the tradeoff between the stopband energy and the deviation of the passband from the ideal filter. The search space is limited to a small number of samples (in the transition band), imposing the condition that the resulting filter have a large number of zeroes in the stopband. Secondly, three different structures to implement the decimation (or interpolation) filter are proposed. The implementation complexity, i.e. the number of multiplications and additions per input sample, are derived for each structure. The results shown that, without taking into account finite word-length effects, the most efficient implementation depends on the filter length to decimation (or interpolation) ratio.

This work was partially supported by the Spanish Ministry of Education and Science under grant PR2007-0218, and by Comunidad Autónoma de Madrid and Universidad de Alcalá through projects CCG07-UAH/TIC-1740, CCG08-UAH/TIC-3941, and CCG08-UAH/TIC-4054.

F. Cruz-Roldán is with the Department of Teoría de la Señal y Comunicaciones, Escuela Politécnica Superior de la Universidad de Alcalá, 28871 Alcalá de Henares (Madrid), SPAIN (phone: + 34 91 885 66 93; fax: + 34 91 885 66 99; e-mail: fernando.cruz@uah.es).

J. D. Osés del Campo and J. I. Godino–Llorente are with the Department of Ingeniería de Circuitos y Sistemas, Escuela Universitaria de Ingeniería Técnica de Telecomunicación, Universidad Politécnica de Madrid, 28031 Madrid, SPAIN (phone: + 34 91 336 7831; fax: + 34 91 336 78 29; e-mail: igodino, doses@ics.upm.es).

L. Boquete-Vázquez is with the Department of Electrónica, Escuela Politécnica Superior de la Universidad de Alcalá, 28871 Alcalá de Henares (Madrid), SPAIN (phone: + 34 91 885 65 72; fax: + 34 91 885 65 91; e-mail: luciano.boquete@uah.es).

C. J. Bleakley is with the School of Computer Science and Informatics, University College Dublin, Belfield, Dublin 4, IRELAND (phone: +353 1 716 2915; fax: +353 1 269 7262; e-mail: chris.bleakley@ucd.ie).

Index Terms

Decimation and interpolation filtering, multirate signal processing, filtering theory, frequency sampling technique.

Polyphase FIR networks based on frequency sampling for multirate DSP applications

I. Introduction

Decimation and interpolation filters (Fig. 1) are used in a wide range of applications, such as data compression, speech enhancement, multi-carrier data transmission, wireless transceivers, digital receivers, software radio receivers, or image size conversion [1]–[9]. In Fig. 1, $P(z)$ is the system function, which is usually a linear-phase low-pass filter, with a cutoff frequency of $\omega_s = \pi/M$ to prevent downsampler-induced aliasing or to remove images after the upsampling process.

One problem these structures have is related to the M value of the decimation or interpolation factor. In some practical systems, such as in decimation schemes for $\Sigma\Delta$ A/D converters or in Software Defined Radio (SDR) receivers, interpolation/decimation ratios can have large values. In these cases, most decimation filter architectures are designed in multistage subfilters, each one obtaining a smaller decimation factor. At the end of the decimation chain, a very selective low-pass Finite Impulse Response (FIR) filter is generally obtained. Besides providing a high value for M, it is also well known that decimation/interpolation filters operating at high rates must be computationally very efficient. This means that high order filters with good selectivity and discrimination are required, and efficient structures to implement them as well.

To address these issues, this paper deals with the design and implementation of these filters using the frequency sampling approach [10], [11]. In the proposed technique, the coefficients are obtained by means of a procedure which only consists of optimizing the values of the samples in the transition band. In this sense, FIR filters with a high length (number of coefficients) can be designed, by optimizing only a small number of samples. In addition, it is well known that the frequency sampling approach is a method that allows recursive implementations of FIR filters. In general, these implementations greatly reduce the number of arithmetic operations in digital filters, especially when most samples are either integers (or powers of 2) or null. Therefore, these structures are beneficial when a low number of frequency samples are nonzero and the length of the prototype filter is large. In this case, efficient implementations from a computational point

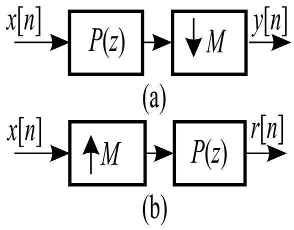

Fig. 1. (a) Decimation and (b) interpolation filters.

of view can be obtained. We address this possibility herein, studying recursive implementations for the polyphase components of the derived filters. For the sake of brevity, we mainly focus our attention on decimation filters, but the results could be extended to interpolation filters.

The rest of this letter is organized as follows. In Section II, we describe a procedure to obtain linear-phase decimation or interpolation filters. The proposed technique is based on a frequency sampling approach which can provide linear-phase filters which also have a very large number of transmission zeros. Section III deals with the implementation structures. First, the relationship between the frequency samples and the M polyphase components is obtained. Certain cases of interest are considered, and a new expression for the polyphase components that allows the system to be implemented with real coefficients is developed. The implementation complexity is studied based on the number of multiplications per input sample (MPIS) and additions per input sample (APIS). The results are compared to conventional polyphase implementation. In Section IV, several examples are included to illustrate the benefits of the proposed technique and to compare the computational cost of different structures. Finally, we summarize our conclusions.

II. Frequency Sampling Technique

The frequency sampling approach is a well-known discrete design technique for obtaining FIR filters [10], [11]. Basically, this technique consists of finding the impulse response coefficients $p\,[n]$ from a desired frequency response specified at a set of equally spaced frequencies; i.e., given the frequency samples $P\,[k]$, the impulse response coefficients are obtained as

$$p\,[n] = \frac{1}{N}\sum_{k=0}^{N-1} P\,[k]\cdot e^{j\frac{2\pi}{N}k\cdot n}, \qquad n = 0,\, 1\cdots, N-1. \tag{1}$$

Expressing the above equation in a matrix form, we get

$$\mathbf{p}^T = \frac{1}{N}\mathbf{W}_N^{-1}\mathbf{P}, \tag{2}$$

where $\mathbf{p} = \left[\begin{array}{cccc} p(0) & p(1) & \cdots & p(N-1) \end{array}\right]$, $\mathbf{W}_N^{-1}$ is the $N \times N$ IDFT matrix, and

$$\mathbf{P}^T = \left[\begin{array}{ccc} P[0] & \cdots & P[N-1] \end{array}\right] \tag{3}$$

contains the desired values of the frequency response.

Deng *et al*. [4] propose an efficient method to design prototype filters for low-delay non-uniform pseudo-Quadrature Mirror Filter banks based on minimizing the objective function

$$\begin{aligned} E = \alpha \tfrac{1}{2\pi} \int\limits_{-\omega_p}^{\omega_p} \left(\left|P\left(e^{j\omega}\right)\right|^2 - \left|P\left(e^{j0}\right)\right|^2 \right) d\omega \\ + (1-\alpha) \tfrac{1}{2\pi} \int\limits_{\omega_s}^{2\pi-\omega_s} \left|P\left(e^{j\omega}\right)\right|^2 d\omega, \end{aligned} \tag{4}$$

where $0 \leq \alpha \leq 1$. With this technique, the filter is designed to minimize a weighted sum of the stopband energy and passband error energy, recording this error as the deviation from the ideal filter in the passband. To formulate the problem, (4) is expressed as

$$E = \mathbf{p} \mathbf{\Phi} \mathbf{p}^{\mathbf{T}} \tag{5}$$

where $\mathbf{\Phi} = \alpha \mathbf{\Phi}_p + (1-\alpha) \mathbf{\Phi}_s$, where the (i,j)-th elements of matrices $\mathbf{\Phi}_p$ and $\mathbf{\Phi}_s$ are obtained as follows [4]:

$$\mathbf{\Phi}_p(i,j) = \begin{cases} 0, & i = j, \\ \frac{1}{\pi}\left(\frac{\sin[\omega_p(i-j)]}{(i-j)} - \omega_p \right), & \text{otherwise}, \end{cases}$$

$$\mathbf{\Phi}_s(i,j) = \begin{cases} 1 - \frac{\omega_s}{\pi}, & i = j, \\ -\frac{\sin[\omega_s(i-j)]}{\pi(i-j)}, & \text{otherwise}. \end{cases}$$

By introducing (2) into (5), the above technique can be reformulated to be used with the frequency sampling approach. Starting from a set of initial frequency samples, the proposed optimization problem is to find a filter that minimizes

$$E = \mathbf{P}^T \mathbf{Q} \mathbf{P},$$

where $\mathbf{Q} = \frac{1}{N^2} \mathbf{W}_N^* \mathbf{\Phi} \mathbf{W}_N^{-1}$.

Note that the maximum value for the stopband cutoff frequency must be $\omega_s = \pi/M$. This means that the maximum number of samples in the passband and transition band is limited to $\lfloor N/(2M) \rfloor + 1$, where $\lfloor \cdot \rfloor$ denotes rounding to the next smaller integer.

Using this technique, if the design process is totally focused on the passband error energy ($\alpha = 1$) or in the stopband energy ($\alpha = 0$), the optimization can cause some problem. In this

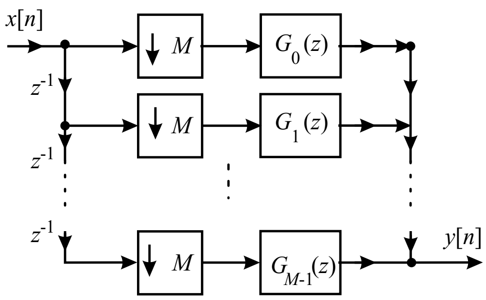

Fig. 2. Type-1 polyphase component structure for the designed filter.

case, as is suggested in [4], the optimization process can be iterative, by fixing an initial value for α and increasing (case of $\alpha_{initial} = 0$) or decreasing (case of $\alpha_{initial} = 1$) it gradually, until an acceptable frequency response is obtained.

III. Implementation Structures

Once the decimation or interpolation filter is designed, the structure to implement it must be chosen. As explained previously, the frequency sampling approach is a method that allows recursive implementation which can greatly reduce the number of arithmetic operations required for digital filtering. Three different structures, therefore, can be employed to give unique implementations, without using multi-stage subfilters decomposition:

1) Through Type-1 (see Fig. 2) or Type-2 polyphase component decomposition [3], using the time-domain filter coefficients. The polyphase filters can be arranged as a parallel realization, in which the output of each filter is selected by a commutator (for more information, see [2]). This structure requires $\lceil N/M \rceil$ multiplications and $(\lceil N/M \rceil - 1)$ additions per input sample [3], where $\lceil \cdot \rceil$ denotes rounding to the next larger integer.
2) For a small number of nonzero frequency samples, the frequency-sampling realization [11] is also an alternative structure for these filters. The system function is characterized by the set of frequency samples as

$$P\left(z\right) = \frac{1 - z^{-N}}{N} \cdot \sum_{k=0}^{N-1} \frac{P\left[k\right]}{1 - e^{j\frac{2\pi}{N}\cdot k} \cdot z^{-1}}. \tag{6}$$

The corresponding filter realization is made up of a cascade of two filters, a comb filter with zeroes located on the unit circle, and a parallel bank of single pole filters also located in identical positions to the zeroes. It is important to note that in practical implementations the structures derived from (6) may have poor finite-word-length behavior, since any uncanceled poles on the unit circle will cause the filter to be unstable. This problem is mitigated by positioning the poles and the zeroes on a circle slightly inside the unit circle, i.e. by

$$P\left(z\right)=\frac{1-r^{N}z^{-N}}{N}\cdot\sum_{k=0}^{N-1}\frac{P\left[k\right]}{1-r\cdot e^{j\frac{2\pi}{N}\cdot k}\cdot z^{-1}}. \tag{7}$$

As a result, the filter is guaranteed to be stable. However, as is claimed in [12], this structure does not completely solve the coefficient quantization, roundoff noise and limit cycles problems. In order to solve these problems, it is important to note that each second-order section is open to multiple implementations. Thus, by carefully choosing the final realization, it is possible to minimize the impact of finite word-length on the performance of the system. For example, structures particularly suitable to operate with poles close to the unit circle can be found in [13], [14], optimal state-space realizations of second-order filters having minimum roundoff noise are described in [15], [16], and procedures to get low sensitivity are presented in [17], etc. It is also possible to apply techniques based on genetic algorithms, to obtain robust second-order structures under finite word-length conditions [18], [19]. In [12], each second-order section is implemented in canonical direct form, using suboptimal and minimum roundoff noise structures, and the performance in terms of noise gain is evaluated.

In our work, in order to obtain the computational complexity, the above problem is not considered, i.e., the study is focused on eq. (7) for $r=1$. In this case, by exploiting the $P\left[k\right]=P^{*}\left[N-k\right]$ symmetry, the recursive structure of Fig. 3 can be used. In this diagram, the coefficients are given by [20]

$$b_{0k}=P\left[k\right]+P\left[N-k\right],$$

$$b_{1k}=P\left[k\right]\cdot e^{-j2\pi k/N}+P\left[N-k\right]\cdot e^{j2\pi k/N},$$

$$a_{1k}=2\cos\left(2\pi k/N\right).$$

If $R+1$ is the number of nonzero samples in the interval $[0,\ \pi]$, this diagram requires the number of MPIS and APIS shown in Table I. It is worth noting that this table shows the

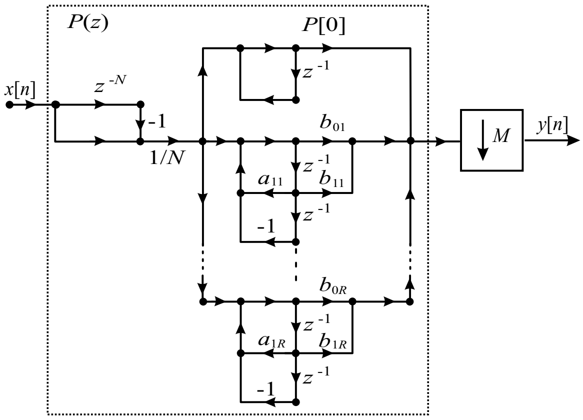

Fig. 3. Frequency sampling structure for the designed filter.

TABLE I

COMPUTATIONAL COMPLEXITY PER INPUT SAMPLE FOR THE STRUCTURES OF FIGS. 3 AND 4.

Structure	MPIS	APIS	Change of Sign
Comb Filter	1	1	1
1st Parallel Filter	1	1	0
Rest of Parallel Filters	$3R$	$3R$	R
Extra	0	1+R	0
Total	2+3R	3+4R	1+R

maximum number of operations, without considering that this number can be reduced in certain cases (for example, usually $P\left[0\right] = 1$). Although a direct implementation of the FIR filter is very inefficient (only one output for every M samples is required at the filter output), the number of operations could be reduced using recursive diagram since it is directly related to the number of nonzero samples.

3) A supposedly more efficient filter structure that introduces the frequency samples into the polyphase components, i.e., recursive implementations of the polyphase components, could be developed. The expressions that describe them, and their implementation complexities, are shown in the following subsections.

The question now is, which is the most efficient structure from a computational point of view, taking the number of MPIS or APIS into account? This point is addressed in the following section. The relationship between the frequency response samples and the polyphase components is obtained. From this relationship, a new structure used to implement the filter bank is derived. For the sake of simplicity, M Type-1 polyphase components are considered, making it extremely easy to apply the results obtained to other kinds of polyphase components.

A. Polyphase FIR Recursive Structures

The M Type-1 polyphase components of the prototype filter can be obtained as

$$G_\ell(z) = \sum_{n=0}^{\lceil N/M \rceil - 1} g_\ell[n] \cdot z^{-n}, \tag{8}$$

where $g_\ell[n] = p[Mn+\ell]$, $\ell = 0,\ 1 \cdots, M-1$. Introducing (1) in (8), the system function $G_\ell(z)$ becomes

$$\begin{aligned} G_\ell(z) &= \sum_{n=0}^{\lceil N/M \rceil - 1} \tfrac{1}{N} \sum_{k=0}^{N-1} P[k] \cdot e^{j\frac{2\pi}{N}k\cdot(Mn+\ell)} \cdot z^{-n} \\ &= \tfrac{1}{N} \sum_{k=0}^{N-1} P[k] \cdot e^{j\frac{2\pi}{N}k\cdot\ell} \sum_{n=0}^{\lceil N/M \rceil - 1} \left(e^{j\frac{2\pi}{N}\cdot M\cdot k} \cdot z^{-1}\right)^n. \end{aligned} \tag{9}$$

Adding the finite geometric series, (9) can be written as

$$G_\ell(z) = \frac{1}{N} \cdot \sum_{k=0}^{N-1} \frac{P[k] \cdot e^{j\frac{2\pi}{N}k\cdot\ell} \cdot \left(1 - e^{j\frac{2\pi}{N}k\cdot M\cdot\left\lceil \frac{N}{M} \right\rceil} \cdot z^{-\left\lceil \frac{N}{M} \right\rceil}\right)}{1 - e^{j\frac{2\pi}{N}\cdot M\cdot k} \cdot z^{-1}}. \tag{10}$$

B. Implementation with Real Coefficients

The prototype filter is usually designed to have real coefficients. Its length can be expressed as $N = mM + s$, where $0 \le s \le (M-1)$. Thus, we get

$$\left\lceil \frac{N}{M} \right\rceil = \begin{cases} m, & \text{for} \quad s = 0, \\ m+1, & \text{for} \quad 1 \le s \le (M-1). \end{cases}$$

If the filter length is $N = mM$, (10) can be simplified as

$$G_\ell(z) = \frac{1 - z^{-\lceil N/M \rceil}}{N} \cdot \sum_{k=0}^{N-1} \frac{P[k] \cdot e^{j\frac{2\pi}{N}k\cdot\ell}}{1 - e^{j\frac{2\pi}{N}\cdot M\cdot k} \cdot z^{-1}}. \tag{11}$$

At this point, it is important to remark that expression (11) will lead to structures with poles and zeroes at the same locations on the unit circle, and those problems derived from the finite precision effects should be taken into account. Again, in order to obtain the computational complexity, we will consider an ideal behaviour.

In general, the frequency samples $P[k]$ are complex, and a direct implementation of (11) requires complex arithmetic. To avoid this complication, the following symmetry conditions can be used: $P[k] = P^*[N-k]$, $e^{j\frac{2\pi}{N}\cdot(N-k)} = e^{-j\frac{2\pi}{N}k}$, which are satisfied for filters with real coefficients. Using the properties described above and combining both the k-th and the $(N-k)$-th single-pole sections, $G_\ell(z)$ can be written in two different ways for $k \neq 0$. For an odd N:

$$\begin{aligned} G_\ell(z) = \tfrac{1-z^{-\lceil N/M \rceil}}{N} \cdot \Big[& \tfrac{P[0]}{1-z^{-1}} \\ & + \sum_{k=0}^{(N-1)/2} \tfrac{\alpha_k + \beta_k \cdot z^{-1}}{1-2\cdot\cos\left(\frac{2\pi}{N}\cdot M \cdot k\right)\cdot z^{-1} + z^{-2}} \Big]. \end{aligned} \tag{12}$$

For an even N:

$$\begin{aligned} G_\ell(z) = \tfrac{1-z^{-\lceil N/M \rceil}}{N} \cdot \Big[& \tfrac{P[0]}{1-z^{-1}} + \tfrac{P[N/2]}{1+z^{-1}} \\ & + \sum_{k=0}^{N/2-1} \tfrac{\alpha_k + \beta_k \cdot z^{-1}}{1-2\cdot\cos\left(\frac{2\pi}{N}\cdot M \cdot k\right)\cdot z^{-1} + z^{-2}} \Big]. \end{aligned} \tag{13}$$

In both eqs. (12) and (13),

$$\alpha_k = 2 \cdot |P[k]| \cdot \cos\left(\frac{2\pi}{N} k \cdot \left(\ell - \frac{N-1}{2}\right)\right) \tag{14}$$

and

$$\beta_k = 2 \cdot |P[k]| \cdot \cos\left(\frac{2\pi}{N} k \cdot \left(\ell - M - \frac{N-1}{2}\right)\right). \tag{15}$$

Considering that $P[N/2] = 0$ for even-length low-pass prototype filters, the second term of (13) is null. Therefore, the realization diagram of Fig. 4, where $a_k = 2 \cdot \cos\left(\frac{2\pi}{N} \cdot M \cdot k\right)$, describes both cases, and only requires real coefficients.

C. Implementation Complexity

In order to obtain the implementation complexity of each polyphase component, only the practical case of real-coefficient low-pass linear-phase prototype filters and $N = 2mM$ are considered. Table I shows the implementation complexity of the structure plotted in Fig. 4, where $R+1$ is the number of nonzero samples in the interval $[0,\ \pi]$. In this case, the polyphase filters can also be implemented using a parallel realization using the commutator model shown in [2]. Here, then, each polyphase filter performs the computations at M times lower rate than with direct implementation.

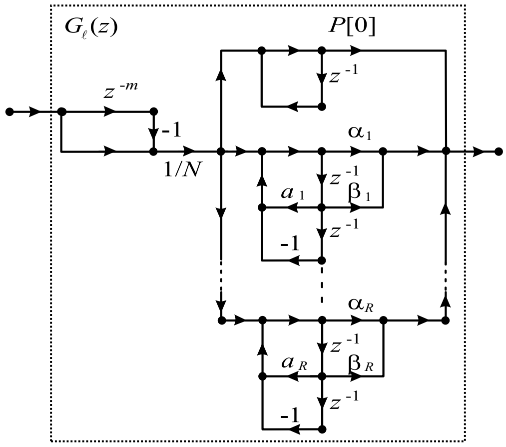

Fig. 4. Polyphase implementation based on the frequency samples. The values of α_k and β_k are given by (14) and (15), respectively.

Thus, the total implementation cost of the polyphase structure based on the frequency samples (Fig. 4) is $3R + 2$ multiplications and $4R + 3$ additions per input sample. Direct polyphase implementation Fig. 2, which also operates at f_s/M, requires $\lceil N/M \rceil$ multiplications and $(\lceil N/M \rceil - 1)$ additions per input sample. Therefore, from a computational point of view and considering the number of multiplications, the implementation of Fig. 4 is more efficient than direct polyphase implementation when $R < (\lceil N/M \rceil - 2)/3$. However, as a main drawback, special care must be taken with regard to the finite precision effects. The above frequency-sampling structures with poles on the unit circle cannot be used in practice due to the noise generated in the loops. In this case, robust structures where the poles have been moved to the inside of the unit circle should be considered to implement the filters [12].

IV. Example Design

In this section, two examples will be provided which can be designed by the proposed technique. Optimization is performed using the `fmincon` function included in the Matlab Optimization Toolbox. The implementation complexity of all the three structures described in this paper will also be shown.

Example 1.- First, the design of a 1050-length linear-phase decimation filter for applications where $M = 105$ is considered. This decimation order can be found in certain receivers for

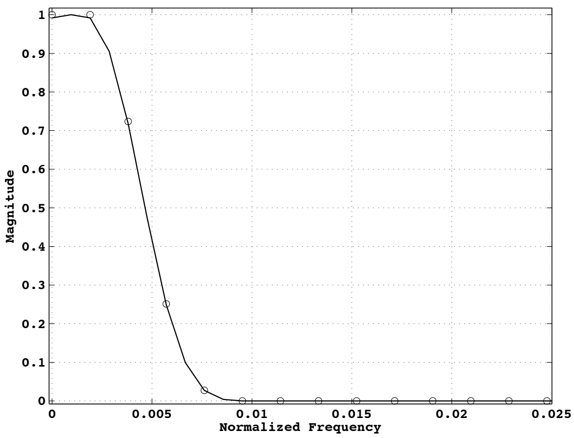

Fig. 5. *Example 1*. Detail of the magnitude response and optimized samples.

TABLE II

Example 1. MAGNITUDE RESPONSE SAMPLES OF THE SYMMETRIC PROTOTYPE FILTER.

k	Initial Values	Optimized $\left\|P\left(e^{jk2\pi/1050}\right)\right\|$
0	1	1
1	0.9045	1
2	0.6545	0.723753832577010
3	0.3455	0.251325117897753
4	0.0955	0.027460652958948
5	0	0.000082949562129
$6 \leq k \leq 525$	0	0

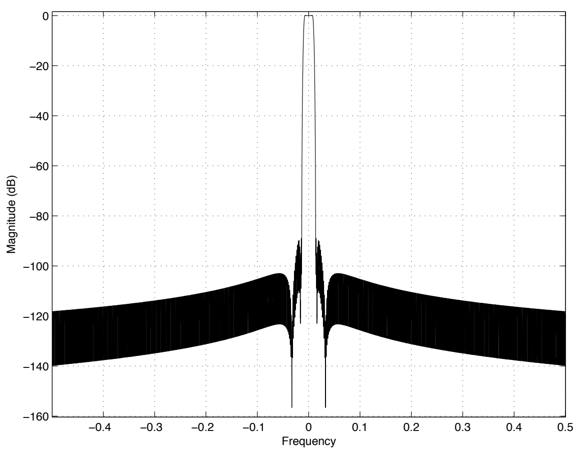

Fig. 6. *Example 2*. Detail of the obtained magnitude response.

software defined radio [5]. We consider in this design that $\alpha = 0.00001$. The initial[1] and optimized values for the frequency response are provided in Table II, and Fig. 5 shows a detail of the resulting magnitude response and the optimized samples. The minimum stopband attenuation for this filter is MSA=-89.9131 dB. The total number of MPIS and APIS for the different structures described in this paper are given in Table III.

Example 2.- A new filter is designed, maintaining the same parameters as in the previous example ($M = 105$, $\alpha = 0.00001$ and $q = 5$), but, in this case, increasing the filter length to $N = 4200$. In this second example, the decimation (interpolation) ratio allows establishment of guard interval, in order to reduce the aliasing (imaging) effects. In this case, a number of samples in the pass and transition bands smaller than the maximum number can be used. The initial and optimized (after running the simulations) values of the frequency response are given in Table IV, and the total number of MPIS and APIS are shown in Table V. Finally, Fig. 6 shows a detail of the resulting magnitude response, with MSA=-102.9096 dB.

In the previous Sections, it was shown that three different structures could be used to implement the decimation (or interpolation) filters. (a) Type-1 polyphase component structure (Fig. 2), in which each polyphase filter is implemented using conventional non-recursive structures (such as direct and transposed forms); (b) the frequency sampling structure shown in Fig. 3; and (c) the recursive structure of Fig. 4 for each polyphase component of Fig. 2. This last structure is useful when there is a large number of null frequency response samples. This condition can be easily fulfilled using the technique proposed in Section II to design the decimation (interpolation) filter.

The structure in Fig. 3 is the most computationally complex one, since $M - 1$ samples are discarded after the filtering stage. From the results obtained in Section III, it can be seen that the structure of Fig. 4 is more computationally efficient when a high N/M ratio and a low number of nonzero samples are used. This situation is typical when high discrimination and selectivity filters with a guard interval are required, in order to reduce the aliasing (imaging) effects which are the result of an adjusted downsampling (upsampling) process. On the other hand, classical Type-1 (Fig. 2) or Type-2 polyphase decompositions implemented by means of a non-recursive structure are more interesting for low N/M ratios. Moreover, these latter structures are always

[1] The initial magnitude response values in the transition band are obtained using a raised cosine frequency characteristic with a rolloff factor $\beta = 1$, i.e., $(1 + \cos(\pi(1 : q)/q))/2$, for $q = 5$

TABLE III

Example 1. COMPUTATIONAL COMPLEXITY PER INPUT SAMPLE

Structure	MPIS	APIS	Frequency Rate
Fig. 2	10	9	f_s/M
Fig. 3	16	23	f_s
Fig. 4	16	23	f_s/M

stable and are more robust when implemented using finite-precision arithmetic.

V. CONCLUSION

A frequency sampling technique, minimizing the energies of the stopband and the error in the pass-band, is described. Since this technique allows implementation of FIR filters by means of non-recursive and recursive structures, three different ways of realizing multirate systems are shown: direct (Fig. 3), parallel polyphase component-based (Fig. 2), and recursive parallel polyphase structures (Fig. 4). The results show that, in general, for low N/M ratios the most efficient structure is the one plotted in Fig. 2. Moreover, this structure presents good properties attending to the effects inherent in using finite-precision arithmetic. For high N/M ratios and also when there are a low number of nonzero samples in the frequency response, the total number of MPIS and APIS can be reduced in the direct structure. In future research, finite word length effects must be considered. Since each recursive second-order section is open to multiple implementations exhibiting different behavior with regard to roundoff noise, coefficient sensitivity, etc. The best structure for the second-order sections must be selected in order to minimize the impact of finite word-length conditions on the final implementation.

ACKNOWLEDGEMENT

The authors would like to thank the Editor-in-Chief and the anonymous reviewers for their valuable comments and constructive criticisms on the manuscript which made this final contribution considerably easier for the reader to grasp.

TABLE IV

Example 2. MAGNITUDE RESPONSE SAMPLES OF THE SYMMETRIC PROTOTYPE FILTER.

k	Initial Values	Optimized $\left\|P\left(e^{jk2\pi/4200}\right)\right\|$
$0 \leq k \leq 4$	1	1
5	0.9045	1
6	0.6545	0.738845199854484
7	0.3455	0.269995641798031
8	0.0955	0.030571896208598
9	0	0.000068724205677
$10 \leq k \leq 2100$	0	0

TABLE V

Example 2. COMPUTATIONAL COMPLEXITY PER INPUT SAMPLE

Structure	MPIS	APIS	Frequency Rate
Fig. 2	40	39	f_s/M
Fig. 3	31	43	f_s
Fig. 4	31	43	f_s/M

REFERENCES

[1] A. N. Akansu and M. J. Medley (Edt.), Wavelet, Subband and Block Transforms in Communications and Multimedia. Kluwer Academic Publishers, Norwell MA, 1999.

[2] R. E. Crochiere and L. R. Rabiner, Multirate Signal Processing. Englewood Cliffs, NJ, Prentice-Hall, 1983.

[3] P. P. Vaidyanathan, Multirate Systems and Filter Banks. Prentice–Hall, Englewood Cliffs NJ, 1993.

[4] Y. Deng, V. J. Mathews, and B. Farhang-Boroujeny, "Low-Delay Nonuniform Pseudo-QMF Banks With Application to Speech Enhancement," IEEE Transactions on Signal Processing, vol. 55, no. 5, 2110-2121, May 2007.

[5] H. Rajić and H. Babić, "Efficient implementation of sharpened CIC decimation filters for software radio receivers," in 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004, vol.3, 5-8 Sep. 2004, 1672- 1676.

[6] M. Laddomada and M. Mondin, "Decimation schemes for $\Sigma\Delta$ A/D converters based on Kaiser and Hamming sharpened filters," IEE Proceedings - Vision, Image and Signal Processing, vol. 151, no. 4, 287- 296, 30 Aug. 2004.

[7] A. Ghazel, L. Naviner, and K. Grati "On design and implementation of a decimation filter for multistandard wireless transceivers," IEEE Transactions on Wireless Communications, vol. 1, no. 4, 558-562, Oct. 2002.

[8] J. T. Kim "Efficient implementation of polynomial interpolation filters or full digital receivers," IEEE Transactions on Consumer Electronics, vol. 51, no. 1, 175-178, Feb. 2005.

[9] S. Yang and T. Q. Nguyen "Interpolated Mth-band filters for image size conversion," IEEE Transactions on Signal Processing, vol. 50, no. 12, 3028-3035, Dec. 2002.

[10] L. R. Rabiner, B. Gold, and C. A. McGonegal, "An approach to the approximation problem for non–recursive digital filters," IEEE Transactions on Audio and Electroacoustics, AU-18, 83-106, Jun. 1970.

[11] J. G. Proakis and D. G. Manolakis, Digital Signal Processing: Principles, Algorithms and Applications. Mac-Millan, New York NY, third edition, 1996.

[12] G. Calvagno, G. Cortelazzo and G. A. Mian, "Minimum noise realization of frequency sampling FIR filters," in IEEE International Symposium on Circuits and Systems, ISCAS 1991, vol. 1, 156-159, Jun. 1991.

[13] N. G. Kingsbury, "Second-order recursive digital-filter element for poles near the unit circle and the real z axis," Electronics Letters, vol. 8, no. 6, 155-156, 23rd March 1972.

[14] D. K. Wise, "The modified Chamberlin and Zölzer filter structures," in Proc. of 9th International Conference on Digital Audio Effects (DAFx-06), Montreal (Canada), September 18-20, 2006, 53-56.

[15] L. B. Jackson, A. G. Lindgren, and Y. Kim, "Optimal synthesis of second-order state-space structures for digital filters," IEEE Transactions on Circuits and Systems, vol. CAS-26, no. 3, 149-153, Mar. 1979.

[16] C. Barnes, "On the design of optimal state-space realizations of second order digital filters," IEEE Transactions on Circuits and Systems, vol. 31, no. 7, 602-608, Jul. 1984.

[17] P. P. Vaidyanathan and S. K. Mitra, "A general theory and synthesis procedure for low sensitivity digital filter structures," Tech. Rep. 82-09, Dep. Elec. Comput. Eng., Univ. of California, Santa Barbara, 1982.

[18] S. P. Harris and E. C. Ifeachor, "Automatic design of frequency sampling filters by hybrid genetic algorithm techniques," IEEE Transactions on Signal Processing, vol. 46, no. 12, 3304-3314, Dec. 1998.

[19] K. Uesaka and M. Kawamata, "Synthesis of low-sensitivity second-order digital filters using genetic programming with automatically defined functions," IEEE Signal Processing Letters, vol. 7, no. 4, 83-85, Apr. 2000.

[20] A. Antoniou, Digital Filters: Analysis, Design, and Applications. New York, NY: Mac-Graw Hill, second edition, 1993.