The Design and Multiplier-Less Realization of Software Radio Receivers With Reduced System Delay

K. S. Yeung and S. C. Chan, *Member, IEEE*

***Abstract*—This paper studies the design and multiplier-less realization of a new software radio receiver (SRR) with reduced system delay. It employs low-delay finite-impulse response (FIR) and digital allpass filters to effectively reduce the system delay of the multistage decimators in SRRs. The optimal least-square and minimax designs of these low-delay FIR and allpass-based filters are formulated as a semidefinite programming (SDP) problem, which allows zero magnitude constraint at $\omega = \pi$ to be incorporated readily as additional linear matrix inequalities (LMIs). By implementing the sampling rate converter (SRC) using a variable digital filter (VDF) immediately after the integer decimators, the needs for an expensive programmable FIR filter in the traditional SRR is avoided. A new method for the optimal minimax design of this VDF-based SRC using SDP is also proposed and compared with traditional weight least squares method. Other implementation issues including the multiplier-less and digital signal processor (DSP) realizations of the SRR and the generation of the clock signal in the SRC are also studied. Design results show that the system delay and implementation complexities (especially in terms of high-speed variable multipliers) of the proposed architecture are considerably reduced as compared with conventional approaches.**

***Index Terms*—Design and multiplier-less realization, low delay, passband linear-phase finite-impulse response (FIR) and allpass filters, sampling rate conversion, semidefinite programming (SDP), software radio receiver (SRR), variable digital filters (VDF).**

I. Introduction

SOFTWARE radio is a general hardware/software platform for supporting inter-communication between different wireless communication systems [1]. The basic idea of an ideal software radio receiver (SRR) is to digitize the received signal using high-speed analog-to-digital converters (ADCs) and to process it by a sophisticated programmable system, probably consisting of a combination of hardware that is re-configurable or programmable [such as field programmable gate array (FPGA)], and digital signal processors (DSPs). Due to various limitations of current digital technology and signal converters, most software radio architectures considered digitize the decimated signal at the intermediate frequency (IF).

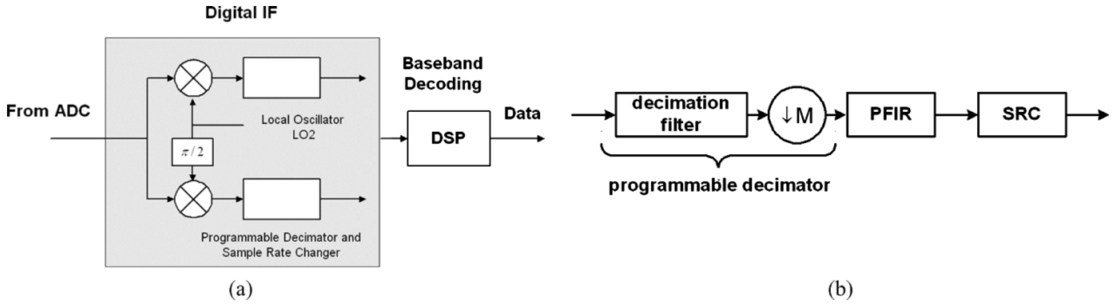
There are several important contributions to the realization of digital IFs of SRRs [1]–[3] (see also the references therein), which is mostly based on the architecture shown in Fig. 1(a). We can see that the analog IF signal is first digitized at a bandwidth of say 20 to 40 MHz. A programmable digital decimator and a sample rate converter (SRC) are employed to isolate the desired user's channel from the signal spectrum and convert it to an appropriate sampling rate for further processing. Moreover, the programmable digital decimator usually consists of multiple stages of decimators to reduce implementation complexity and power dissipation as shown in Fig. 1(b). As the sampling rate of the baseband signal is much lower than that at the IF, the output of each stage in the decimator will consist of a bandlimiting (anti-aliasing) digital filter and a downsampler (decimator) to filter out the unwanted signals and lower the sampling rate. By selecting an appropriate number of stages, different integer downsampling ratios can be implemented. The programmable FIR filter (PFIR) is used to remove the residual interference from adjacent channels. It is because the sampling rate is usually not an integer multiples of the channel spacing. Hence, the multistage decimators, which implement an integer downsampling ratio, are unable to remove this residual interference. Together with the SRC, which provides the necessary arbitrary rate-change factor, it is now possible to accommodate signals with a wide variety of bandwidths.

One drawback of this conventional structure is that the output of the multistage decimators, which is obtained by downsampling the high-rate IF signal from the ADC, has to be upsampled again in order to carry out the arbitrary sample rate conversion. Another important problem is the high complexity of the PFIR due to a considerable number of high-speed variable multipliers required for its implementation, especially for wideband signals. Recently, the authors have proposed a new digital IF architecture for SRRs shown in Fig. 2 [4], [5]. The SRC, which is realized using a Farrow-based variable digital filter (VDF) [6], [7], is performed immediately after the multistage decimators. The basic idea of the VDF-based SRC is to provide variable fractional delay in the passband and additional attenuation in the stopband. This allows us to replace the PFIR by a HBF with fixed coefficients, if the arbitrary rate-change factor is properly chosen. This new architecture eliminates the need for the PFIR, which is usually a bottleneck of software radio application for wideband signals. As a result, the implementation complexity is significantly reduced because the fixed coefficients of the SRR can be efficiently implemented using sum-of-powers-of-two (SOPOT) coefficients or canonical signed digit

Manuscript received July 26, 2003; revised May 7, 2004. This paper was presented in part at the International Symposium on Circuits and Systems, Bangkok, Thailand, in May 2003. This paper was recommended by Associate Editor J. McCanny.

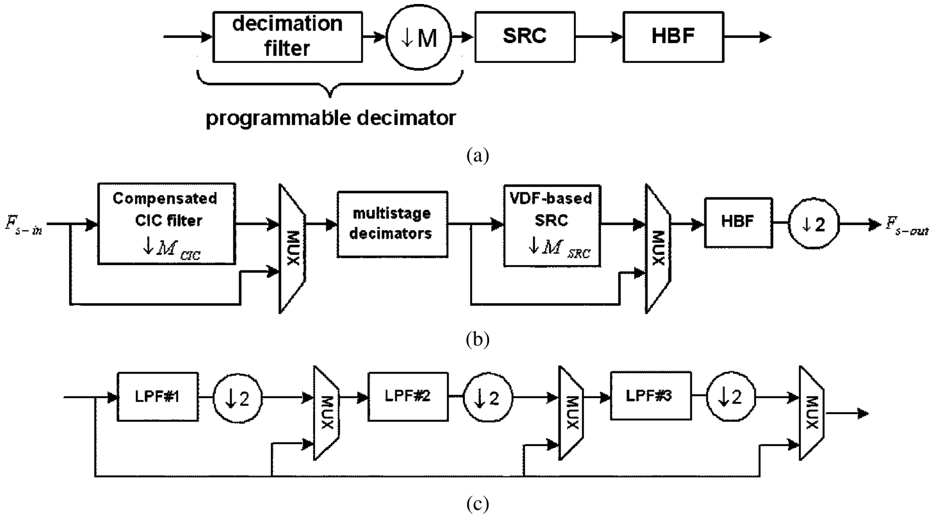
The authors are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong (e-mail: scchan@eee.hku.hk).

Digital Object Identifier 10.1109/TCSI.2004.838253



(a) (b)

Fig. 1. (a) Digital IF architecture for SRR. (b) Conventional programmable decimator and SRC.



(a)

(b)

(c)

Fig. 2. (a) Proposed architecture of the programmable decimator and SRC. (b) Architecture of the proposed SRR. (c) Architecture of the multistage decimators.

(CSD) [8]. Apart from the limited number of variable multipliers required in the Farrow structure, the entire SRR can be implemented without any multiplications.

In [4], the multistage decimators and the HBF are realized using linear-phase FIR filters. The use of linear-phase filters usually results in a longer system delay compared with approximately passband linear-phase (low-delay) FIR or IIR filters. This is undesirable in some applications and it motivates us to study in this paper the application of low-delay FIR, digital allpass filters and SRC and their efficient realizations in order to reduce the system delay of the new SRR. The design of the low-delay FIR and allpass-based filters are performed using semidefinite programming (SDP) [9]–[11]. Furthermore, it was found for the low-delay FIR decimators that the constraint of zero magnitude response at $\omega = \pi$, which is desirable to attenuate the aliasing components before decimation, can be readily incorporated in the SDP approach. To design the allpass filters using SDP, the frequency specification is first formulated as a set of matrix inequalities, which is a bilinear function of the filter coefficients and the ripple to be minimized. The overall design problem turns out to be a quasiconvex constrained optimization problem and it can be solved through a series of convex optimization sub-problems and the bisection search algorithm [9]. Besides, the design of the VDF-based SRC is further studied. This SRC is also applicable to software radio transmitters and base stations [12], as they also require arbitrary sampling rate conversion. In particular, a new SDP method, which is optimal in the minimax design criterion, is proposed and compared with the weight least squares (WLS) method [6], [7]. Design results show that both methods give similar performances when the order of interpolation is small and the computational time of WLS is significantly lower. If higher order interpolation and additional constraints are required, the SDP method is more flexible and it yields better results, at the expense of increased design time. Other implementation issues of the SRC, such as the generation of the clock signal and control parameters, are also investigated. In particular, a flexible clocking generation scheme to accommodate different communication standards and a unit to calculate the control parameter in the interpolation part of the SRC are proposed.

As mentioned earlier, another objective of this paper is to study the efficient realization of the proposed low-delay receivers. Two approaches are considered. The first one is to realize the SRR using DSP. It is assumed that the DSP is fast enough to deal with the decimated signal in the SRR and its output will be further processed in the baseband, which is more suitable for software radio applications with large downsampling ratios. The second one is the multiplier-less hardware realization, which is more desirable when the downsampling ratio is small, i.e., high rate operations. In the latter approach, the fixed coefficients of the SRR can be efficiently implemented as limited number of shifts and additions by employing the SOPOT representations. These SOPOT coefficients are obtained by the random search algorithm in [4], [13]. The multiplier-block (MB) technique [14] is also employed to further reduce the implementation complexity. Design results show that the complexity of the proposed allpass-based SRR is less than that using the low-delay FIR filters with the same design specifications for both DSP and multiplier-less implementations. Both approaches compare favorably with the conventional approach in terms of the number of general multipliers required and system delay as demonstrated by a design example for a multistandard receiver for the GSM, W-CDMA, CDMA2000, and Hiperlan/2 wireless interfaces. A reduction in system delay ranging from 10.4% to 15% is achieved over their linear-phase counterpart in [4], at the expense of modest increase in arithmetic complexity. It should be noted that the proposed techniques for realizing the low-delay decimators are also applicable to conventional receivers.

The rest of this paper is organized as follows: Section II is devoted to the design and implementation of the proposed low-delay FIR and allpass-based SRR. Comparisons and detailed design examples for different communication standards are illustrated in Section III. Section IV describes other implementation issues of the SRC for the SRR. Finally, conclusions are drawn in Section V.

II. Proposed Low-Delay SRRs

In this section, the design and implementation of the proposed low-delay FIR and allpass-based digital IF architecture for SRRs are described. Fig. 2 shows the new digital IF architecture proposed in [4], [5]. In Fig. 2(b), the digitized IF-signal from the high-speed ADC is first passed through the compensated cascaded integrator-comb (CIC) filter and is decimated by a factor of M_{CIC}. Its output is then fed to the multistage decimators, which are realized using general low-pass anti-aliasing filters, denoted by LPF#1, LPF#2, and LPF#3 in Fig. 2(c). The number of low-pass anti-aliasing filters required depends on the maximum downsampling ratio of the receiver. Without loss of generality, we assume that our receiver consists of three stages so that they can support the signal bandwidths ranging from GSM to Hiperlan/2 standards (i.e., a downsampling ratio from 4 to 295.3849. The maximum downsampling ratio of the SRR is 512). The maximum downsampling ratio can be increased, to say 1024 and higher, by increasing the number of anti-aliasing filters and the decimation factor of the CIC filter. Unlike the conventional receivers in [1]–[3], the output of the multistage decimators is fed to the SRC, which is implemented using a Farrow-based VDF. Finally, the output of the VDF-based SRC is fed to a half-band filter (HBF) with fixed coefficients to reduce the residual interference. An advantage of this architecture is that it eliminates the need for a PFIR in the traditional receiver, which is usually a bottleneck in software radio application for wideband signals. The overall downsampling ratio M_{SRR} of the proposed SRR is given by

$$M_{\text{SRR}} = M_{\text{CIC}} \cdot M_{\text{SRC}} \cdot 2^k \tag{2.1}$$

where M_{CIC}, which is a positive powers-of-two integer, is the downsampling ratio of the compensated CIC filter; M_{SRC}, which is chosen to lie between 1 and 2, is the arbitrary downsampling ratio of the SRC; and $k \in \{1,2,3,4\}$ is the number of the remaining 2-to-1 decimators to be selected. In general, the VDF-based SRC is more complicated and involved to design and realize than the other digital filters in the SRR. Therefore, it is preferable to implement the SRC after the compensated CIC filter and the multistage decimators so that the operating rate of the SRC can be lowered. The system delay G_{SRR} of the proposed SRR with $k \in \{1,2,3,4\}$ given by

$$\begin{aligned} G_{\text{SRR}} = & G_{\text{CIC}} + M_{\text{CIC}} \\ & \times \big[G_{\text{second}} + G_{\text{LPF1}} + G_{\text{LPF2}} \cdot 2^{k-3} \\ & + G_{\text{LPF3}} \cdot 2^{k-2} + G_{\text{SRC}}(\phi) \cdot 2^{k-1} \\ & + G_{\text{HBF}} \cdot M_{\text{SRC}} \cdot 2^{k-1}\big] \end{aligned} \tag{2.2}$$

where $G_{\text{CIC}} = (M_{\text{CIC}} - 1)L/2$ is the group delay of the CIC filter; G_{second} is the group delay of the second-order CIC compensator to be described later in Section II-A; G_{LPF1}, G_{LPF2}, and G_{LPF3} are the group delays of the LPF#1, LPF#2, and LPF#3, respectively; $G_{\text{SRC}}(\phi)$ is the group delay of the SRC as a function of ϕ; G_{HBF} is the group delay of the HBF. Note that if one of the decimation filters is not selected, the corresponding group delay should be zero. It can be seen that the system delay mainly depends on the group delay of the LPFs and HBF since they increase rapidly with the downsampling ratios of M_{CIC}, M_{SRC} and k. As a result, if low-delay FIR or allpass filters are used to realize the multistage decimators and HBF, then the system delay can be greatly reduced. Next, let us go through the architecture of the proposed SRR in detail. The techniques to be described in Sections II-A–C are also applicable to traditional receivers [2], [3] though our primary interest will be the architecture proposed in [4], [5].

A. Second-Order CIC Compensator

Here, the design and implementation of the second-order CIC compensator to compensate for the passband droop of the basic CIC filter are described. The basic CIC filter [15] is commonly employed when a large downsampling ratio is required, because of its reasonable performance and low hardware complexity. The transfer function of the CIC filter is given by

$$H_{\text{CIC}}(z) = \left[\frac{H_C(z^{M_{\text{CIC}}})}{M_{\text{CIC}} \cdot H_C(z)}\right]^{L_{\text{CIC}}} \tag{2.3}$$

where $H_C(z) = 1 - z^{-1}$; L_{CIC} is the number of CIC stages. One drawback of the CIC filter is the passband droop that limits the quality of the anti-aliasing filters. In [5], we proposed a second-order CIC compensator with the following transfer function:

$$P(z) = a + bz^{-1} + az^{-2} \tag{2.4}$$

where a and b are real-valued constants to be determined and $G_{\text{second}} = 1$. As shown in Fig. 3(a), it is placed after the CIC filter. This compensator can also be viewed as the equalizer in the interpolated FIR filters [16]. Its frequency response, as can be seen from Fig. 4, is periodic, which is designed to equalize the passband droop of the CIC filter. Given the frequency response of the CIC filter in (2.3), the constants a and b can be readily determined using the Parks–McClellan algorithm. To reduce the implementation complexity, the constants a and b are expressed as the following CSD or SOPOT representations [8]:

$$\hat{b} = \sum_{r=1}^{R} u_r 2^{a_r} \tag{2.5}$$

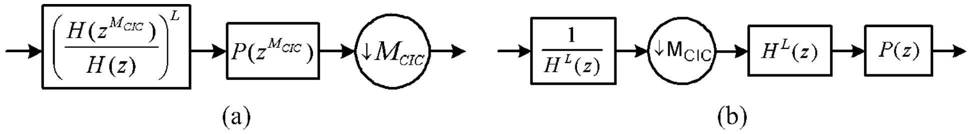


Fig. 3. Compensated CIC filter: (a) before and (b) after the application of the noble identity. (Note: the scaling of $(1/M_{\mathrm{CIC}})^{L_{\mathrm{CIC}}}$ is not shown.)

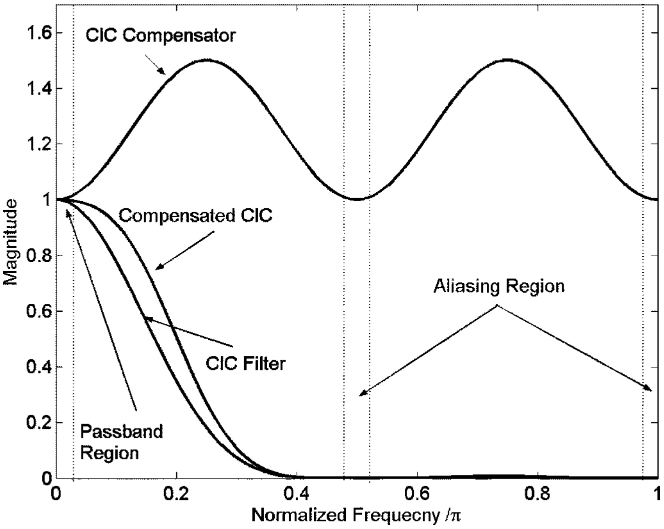


Fig. 4. Frequency responses of the CIC filter, CIC compensator and compensated CIC filter ($M_{\mathrm{CIC}} = 4$ and $L = 4$).

where $u_r \in \{-1, 1\}$ and $a_r \in \{-l_b, \ldots, -1, 0, 1, \ldots, u_b\}$; l_b and u_b are positive integers and their values determine the dynamic range of the coefficients; R is the number of terms used in the coefficient approximation. Using (2.5), the coefficient multiplications can be efficiently implemented as limited number of shifts and additions only. These SOPOT coefficients can be obtained by a number of methods [4], [8], [13], [17]–[19]. In this paper, the random search algorithm [4], [13] is employed to minimize the total number of SOPOT terms subject to the given specifications in the frequency domain. The resulting SOPOT coefficients are: $a = -2^{-3}$; $b = 2^0 + 2^{-2}$. The frequency responses of the basic CIC filter, the CIC compensator, and the compensated CIC filter for $M_{\mathrm{CIC}} = 4$ and $L_{\mathrm{CIC}} = 4$ are shown in Fig. 4. The worst case passband deviation and aliasing attenuation of the compensated CIC filter for $M_{\mathrm{CIC}} \geq 2$ and $L_{\mathrm{CIC}} = 4$ are 0.0085 and 112.34 dB, while those for the CIC filter are 0.0338 and 112.36 dB, respectively. Therefore, the CIC compensator improves the passband droop by a factor of four while maintaining a comparable aliasing attenuation. It also has a low coefficient dynamic range compared with the interpolated second-order polynomial (ISOP) filters in [2]. Using the noble identity [20], the compensated CIC filter in Fig. 3(a) can be implemented more efficiently as shown in Fig. 3(b), and the structure of the basic CIC filter is shown in Fig. 5. Next, we shall consider the design and implementation of the low-delay low-pass anti-aliasing filters in the multistage decimators.

B. Multistage Decimators

1) Design: Conventionally, the low-pass anti-aliasing filters are implemented using halfband filters [2], [3]. In [4], linear-phase FIR low-pass anti-aliasing filters (LPFs): LPF#1, LPF#2, and LPF#3, as shown in Fig. 2(c), are proposed to improve the performance of the SRR. Their coefficients are readily obtained using the Parks–McClellan algorithm and efficiently implemented using SOPOT coefficients and MB. As mentioned earlier, this approach usually yields longer system delay. In this paper, we propose to realize these multistage decimators using low-delay FIR and digital allpass filters in order to reduce the system delay of the SRR. First of all, let us consider the design of the allpass-based decimation filters with the following transfer function

$$A_N(z) = \frac{z^{-N} \cdot T_N(z^{-1})}{T_N(z)} \tag{2.6}$$

where N is the filter order; $T_N(z) = \sum_{n=0}^{N} a_n z^{-n}$ with $a_0 = 1$, and a_n's are real-valued coefficients. Substituting $z = e^{j\omega}$ into (2.6), we have $A_N(e^{j\omega}) = e^{-jN\omega} \cdot e^{2j\phi_A(\omega)} = e^{j\theta(\omega)}$, where $\theta(\omega) = -N\omega + 2\phi_A(\omega)$, and $\tan(\phi_A(\omega)) = (\sum_{n=0}^{N} a_n \cdot \sin(n\omega) / \sum_{n=0}^{N} a_n \cdot cos(n\omega))$. It can be seen that the allpass filter has a unit magnitude response and its phase response can be used to approximate a desired phase response. Here, it is used to realize the low-pass anti-aliasing filters in the multistage decimators as a parallel interconnection of two allpass sections, Fig. 6(a), as follows:

$$H_{\mathrm{LPFi}}(z) = \frac{\left[z^{-(N_i-1)} + A_{N_i}(z)\right]}{2}, \qquad i = \{1, 2, 3\}. \tag{2.7}$$

Since fractional delays are, in general, not required in the multistage decimators, one of the allpass sections is chosen as a signal delay in order to reduce the implementation complexity. The desired phase response of the allpass filter for the low-pass filters is given by

$$\theta_i^{(d)}(\omega) = \begin{cases} -(N_i-1)\omega, & 0 < |\omega| < \omega_{p,i} \\ -(N_i-1)\omega - \pi, & \omega_{s,i} < |\omega| < \pi \end{cases}, \quad i = \{1, 2, 3\} \tag{2.8}$$

where $\omega_{p,i}$ and $\omega_{s,i}$ are the passband and stopband edges of the ith low-pass filter in the multistage decimators. It should be noted that a zero at $\omega = \pi$ is structurally imposed in $H_{\mathrm{LPFi}}(z)$, which is desirable to attenuate the aliasing components. A number of methods have been proposed for designing allpass filters [21]–[24]. In this work, the SDP approach [9] is employed. This approach is able to design causal-stable digital allpass filters with a prescribed pole radius constraint and minimax design criterion. Additional linear constraints such as flatness or zero magnitude response at certain frequencies can be incorporated. Interested readers are referred to [9] for more details.

For the design of low-delay FIR decimators, the SDP approach in [11] is also employed. This is because it is possible to incorporate the zeros at $\omega = \pi$ for the low-delay anti-aliasing filters. Traditionally, linearly constrained linear-phase FIR filters are designed using a linear programming approach [25].

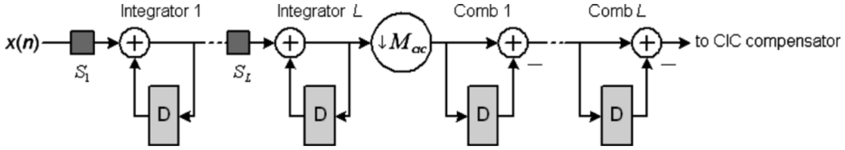


Fig. 5. Architecture of an L_{CIC}-stage CIC decimation filter. S_k: programmable shifter.

To the authors' best knowledge, the optimal minimax design of nonlinear-phase FIR filters with linear constraints and convex quadratic constraints has not been reported. Next, we show that this problem can be solved readily using SDP. More precisely, the ith low-pass filter in the multistage decimators of length N_i to be designed is given by

$$H_i(z) = \sum_{n=0}^{N_i-1} h_i(n) z^{-n} = z^{-(N_i-1)} \tilde{H}_i(z) \qquad (2.9)$$

where $\widetilde{H}_i(z) = \sum_{n=0}^{N_i-1} h_i(n) z^{-(N_i-1-n)}$, with the following desired frequency response:

$$H_i^{(d)}(e^{j\omega}) = \begin{cases} e^{-j\tau_i\omega}, & 0 < |\omega| < \omega_{p,i} \\ 0, & \omega_{s,i} < |\omega| < \pi \end{cases}, \qquad i = \{1,2,3\} \qquad (2.10)$$

where τ_i is the corresponding group delay. This will reduce to the linear-phase case when $\tau_i = (N_i - 1)/2$. Let M be the number of zeros to be imposed at $\omega = \pi$ for $H_i(z)$. This is equivalent to

$$\left. \frac{d^m}{d\omega^m} H_i(e^{j\omega}) \right|_{\omega=\pi} = \left. \frac{d^m}{d\omega^m} \tilde{H}_i(e^{j\omega}) \right|_{\omega=\pi} = 0, \qquad \text{for} \quad m = 0, \ldots, M-1. \quad (2.11)$$

Expanding (2.11) and after slight manipulation, one gets a set of linear equality constraints as follows:

$$\boldsymbol{A_i} \cdot \boldsymbol{h_i} = \boldsymbol{B_i} \qquad (2.12)$$

where $[\boldsymbol{A_i}]_{m,n} = ((N_i - 1 - n)!/(N_i - 1 - n - m)!)(-1)^{(N_i-1-n-m)}$, $[\boldsymbol{h_i}]_n = h_i(n)$, and $[\boldsymbol{B_i}]_m = 0$. Here, $[\boldsymbol{A}]_{m,n}$ denotes the (m,n)-th entry of matrix $\boldsymbol{A}$. This will be used to eliminate the redundant variables in the SDP method to be described later in this section. To minimize the maximum ripple of the approximation error is equivalent to the following:

$$\min_{\boldsymbol{h}_i} \max W(\omega) \left| \boldsymbol{h_i^T} \cdot \boldsymbol{e_i} - H_i^{(d)}(e^{j\omega}) \right|, \qquad \text{for } \omega \in [0, \omega_{p,i}] \text{ and } [\omega_{s,i}, \pi] \quad (2.13)$$

where $\boldsymbol{e_i} = [1\ e^{-j\omega}\ \cdots\ e^{-j(N_i-1)\omega}]^T$; $W(\omega)$ is a positive weighting function. To solve (2.13) using SDP, we densely discretize ω over the band of interest into a set of frequency points $\omega_j, j = 1, \ldots, J$. This yields

$$\min_{\boldsymbol{h}_i} \ \delta_i \text{ subject to } \alpha_{R,i}^2(\omega_j) + \alpha_{I,i}^2(\omega_j) \le \delta_i \qquad (2.14a)$$

where $\alpha_{R,i}(\omega_j) = W(\omega_j)|\boldsymbol{h_i^T} \cdot \boldsymbol{c_i} - \mathrm{Re}\{H_i^{(d)}(e^{j\omega_j})\}|$; $\boldsymbol{c_i} = [1\ \cos(\omega_j)\ \cdots\ \cos((N_i-1)\omega_j)]^T$; $\alpha_{I,i}(\omega_j) = W(\omega_j)|\boldsymbol{h_i^T} \cdot \boldsymbol{s_i} + \mathrm{Im}\{H_i^{(d)}(e^{j\omega_j})\}|$; $\boldsymbol{s_i} = [0\ \sin(\omega_j)\ \cdots\ \sin((N_i-1)\omega_j)]^T$. Using the Schur complement [10], it can be shown that (2.14a) is equivalent to

$$\min_{\boldsymbol{h}_i} \ \delta_i \text{ subject to } \boldsymbol{\Gamma}_j(\boldsymbol{h_i}) \ge 0 \qquad (2.14b)$$

where $\boldsymbol{\Gamma}_j(\boldsymbol{h_i}) = \begin{pmatrix} \delta_i & \alpha_{R,i}(\omega_j) & \alpha_{I,i}(\omega_j) \\ \alpha_{R,i}(\omega_j) & 1 & 0 \\ \alpha_{I,i}(\omega_j) & 0 & 1 \end{pmatrix}$, and $\boldsymbol{A} \ge 0$ means that matrix $\boldsymbol{A}$ is positive semidefinite. Since $\boldsymbol{\Gamma}_j(\boldsymbol{h_i})$ is affine in $\boldsymbol{h_i}$, it is equivalent to a set of linear matrix inequalities (LMIs) [10]. In order to simultaneously solve the SDP problem in (2.14b) and the constraint in (2.12), the dependent variables can be expressed as a linear combination of independent variables. The number of variables to be optimized is therefore reduced. It not only speeds up the optimization process but also structurally imposes the desired constraint. To remove the redundant variables, let r_i be the number of redundant variables in $H_i(z)$. (2.12) can be rewritten as

$$[\boldsymbol{A_{i,N_i-r_i}} \quad \boldsymbol{A_{i,r_i}}] \cdot \begin{bmatrix} \boldsymbol{h_{i,N_i-r_i}} \\ \boldsymbol{h_{i,r_i}} \end{bmatrix} = \boldsymbol{B_i} \qquad (2.15)$$

where $\boldsymbol{A_i} = [\boldsymbol{A_{i,N_i-r_i}}\ \boldsymbol{A_{i,r_i}}]$; $\boldsymbol{h_i} = \left[\begin{smallmatrix} \boldsymbol{h_{i,N_i-r_i}} \\ \boldsymbol{h_{i,r_i}} \end{smallmatrix}\right]$. Using (2.15), $\boldsymbol{h_i}$ can be rewritten in terms of $\boldsymbol{h_{i,N_i-r_i}}$ as

$$\boldsymbol{h_i} = \begin{bmatrix} \boldsymbol{O_{N_i-r_i}} \\ \boldsymbol{A_{i,r_i}^{-1} B_i} \end{bmatrix} + \begin{bmatrix} \boldsymbol{I_{N_i-r_i}} \\ -\boldsymbol{A_{i,r_i}^{-1} A_{i,N_i-r_i}} \end{bmatrix} \boldsymbol{h_{i,N_i-r_i}} \qquad (2.16)$$

where $\boldsymbol{O_N}$ is an $(N \times 1)$ null vector; $\boldsymbol{I_N}$ is an $(N \times N)$ identity matrix. Substituting (2.16) into (2.14b) and defining the augmented variable $\boldsymbol{x_i^T} = [\delta_i\ \boldsymbol{h_{i,N_i-r_i}^T}]$, the optimization problem in (2.14b) can be cast into the following standard SDP problem:

$$\min_{\boldsymbol{x}_i} \ \boldsymbol{u_i^T} \cdot \boldsymbol{x_i} \text{ subject to } \boldsymbol{F}(\boldsymbol{x_i}) \ge 0 \qquad (2.17)$$

where $\boldsymbol{u_i^T} = [1\ \boldsymbol{O_{N_i-r_i}^T}]$; $\boldsymbol{F}(\boldsymbol{x_i}) = diag(\boldsymbol{\Gamma}_1(\boldsymbol{x_i}), \ldots, \boldsymbol{\Gamma}_J(\boldsymbol{x_i}))$. Theoretically, it is possible to determine whether a feasible solution exists for the SDP problem, and if so, it is possible to determine the global optimal solution, since the problem is convex. Moreover, the SDP problem is very general in that other design criteria such as least squares, and least squares with peak error constraints can be employed, possibly with linear and convex quadratic constraints. Due to page limitations, their illustrations are omitted.

2) Multiplier-Less Realization: As mentioned earlier, the fixed coefficients of the multistage decimators can be efficiently implemented without multiplications using SOPOT coefficients [8] and the MB technique [14], [26]. When applying the MB technique to the realization of digital infinite-impulse response (IIR) filters, Dempster and Macleod [26] reported that the cascade structure is in general more efficient. Therefore, the allpass filters in our decimators are implemented using a cascade of first- and second-order sections [20] as shown in Fig. 6. More precisely, let r_n be the nth root of $T_N(z)$, $n = 1, 2, \ldots, N$, in (2.6). For real-valued r_n, the first-order section has the form

$$B_k^{(1)}(z) = \frac{b_{k,1} + z^{-1}}{1 + b_{k,1} z^{-1}}, \quad \text{for } k = 1, 2, \ldots, K_1 \qquad (2.18)$$

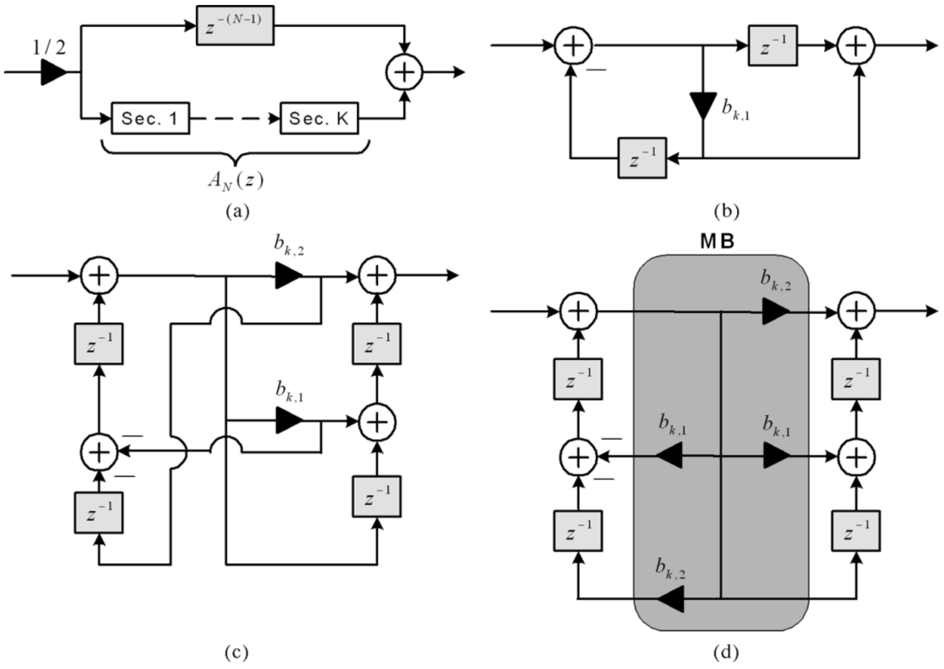


Fig. 6. Implementation of the multistage decimators (LPFs). (a) Parallel interconnection of two allpass sections for LPFs. (b) Structure of the first-order section. (c) and (d) structure of the second-order section for DSP and multiplier-less implementations, respectively.

TABLE I
DESIGN SPECIFICATIONS AND RESULTS OF LPFS, S VDF-BASED SRC, AND HBF USING ALLPASS-BASED FILTERS (LOW-DELAY FIR FILTERS) WITH FIXED SOPOT COEFFICIENTS

	LPF#1	LPF#2	LPF#3	VDF	HBF
Passband edge ω_p	0.05π	0.1π	0.2π	0.4π	0.4π
Stopband edge ω_s	0.925π	0.85π	0.7π	0.7π	0.6π
Group delay (samples)	3	5	10	19 – 20	25
Filter order	4 (8)	6 (13)	11 (22)	39	13 (30)
Number of subfilters (L)	N/A	N/A	N/A	4	N/A
Maximum wordlength in fractional part (bits)	14 (18)	16 (19)	17 (20)	20 (20)	18 (22)
Passband deviation (in *dB*)	5×10^{-7} (2.65×10^{-4})	1.07×10^{-6} (9.25×10^{-5})	1.73×10^{-6} (7.87×10^{-5})	0.0105 (0.0105)	3.12×10^{-10} (8.18×10^{-5})
Stopband attenuation (in *dB*)	103.79 (101.09)	105.62 (100.79)	98.05 (100.15)	95.77 (93.98)	101.44 (100.03)
Group delay error (in *dB*)	-52.39 (-63.09)	-54.89 (-70.46)	-53.98 (-60.91)	-50.17 (-50.17)	-67.96 (-61.94)
Average SOPOT terms per coefficient	4.25 (4.67)	5.5 (5.14)	6.64 (5.09)	4.78 (4.76)	6.46 (5.84)
Adders required for SOPOT coefficients before MB	13 (33)	27 (58)	62 (94)	296 (299)	71 (150)
Adders required for SOPOT coefficients after MB	12 (24)	26 (36)	57 (54)	112 (109)	61 (88)

where $b_{k,1} = -r_n$. When r_n is complex, the corresponding second-order section is

$$B_k^{(2)}(z) = \frac{b_{k,2} + b_{k,1}z^{-1} + z^{-2}}{1 + b_{k,1}z^{-1} + b_{k,2}z^{-2}}, \quad \text{for } k = 1, 2, \ldots K_2 \tag{2.19}$$

where $b_{k,1} = -2\mathrm{Re}\{r_n\}$ and $b_{k,2} = |r_n|^2$; K_1 and K_2 are the number of first- and second-order sections, respectively. The total number of sections in the allpass function $A_N(z)$ is $K_1 + K_2 = K$. For DSP implementation, the multiplications in the first- and second-order sections are implemented by a dedicated high-speed multiplier as shown in Fig. 6(b) and (c), respectively. For multiplier-less (hardware) implementation, the fixed coefficients of the first- and second-order sections can be represented using SOPOT coefficients. For the second-order section, the MB technique is employed to further reduce the complexity as shown in Fig. 6(d). The specifications and performances of the LPFs in the multistage decimators are summarized in Table I. The SOPOT coefficients of the designed LPFs are shown in Tables II–IV. The frequency responses and the corresponding group delays of the allpass-based LPFs are shown in Fig. 7.

TABLE II
SOPOT COEFFICIENTS OF ALLPASS FILTER IN LPF#1
(Filter Order $= 4$, $K_1 = 2$, $K_2 = 1$)

SOPOT Coefficients	SOPOT Coefficients
$b_{1,1} = -2^{-2} + 2^{-14}$	$b_{3,1} = -2^{-7} - 2^{-9} - 2^{-10} - 2^{-12} - 2^{-14}$
$b_{2,1} = 2^{-2} + 2^{-10} + 2^{-11} - 2^{-14}$	$b_{3,2} = 2^{-1} - 2^{-5} - 2^{-8} - 2^{-9} + 2^{-13} + 2^{-14}$

TABLE III
SOPOT COEFFICIENTS OF ALLPASS FILTER IN LPF#2
(Filter Order $= 6$, $K_1 = 0$, $K_2 = 3$)

SOPOT Coefficients	SOPOT Coefficients
$b_{1,1} = -2^{-1} - 2^{-5} - 2^{-11} + 2^{-14}$	$b_{2,2} = 2^{-4} + 2^{-5} + 2^{-8} - 2^{-11} - 2^{-13} - 2^{-16}$
$b_{1,2} = 2^{-4} + 2^{-6} + 2^{-7} + 2^{-9} + 2^{-11} - 2^{-15}$	$b_{3,1} = -2^{-4} + 2^{-8} + 2^{-11} + 2^{-13} - 2^{-16}$
$b_{2,1} = 2^{-1} + 2^{-5} + 2^{-6} + 2^{-9} + 2^{-12} + 2^{-15}$	$b_{3,2} = 2^{-1} + 2^{-5} + 2^{-7} - 2^{-10} + 2^{-15} + 2^{-16}$

Fig. 8(a) shows the pole-zero plots of the LPF#3. It can be seen that all poles of the filter are inside the unit circle. Due to page limitation, pole-zero plots of LPF#1 and LPF#2 and details of the low-delay FIR decimators are omitted. The multiplier-less realization follows closely the approach presented in [4]. Their performance comparison will be presented in Section III. Next, we shall consider the design and implementation of the sample rate converter for arbitrary sample rate conversion.

C. SRC

The design of programmable SRCs with arbitrary conversion factors was studied in detail by Ramstad [27]. In general, there are two approaches to implement a SRC with different tradeoff between the operating rate and the hardware complexity for SRRs. One is to employ the structure in Fig. 9(a) [28] where the input signal is first up-sampled by a factor of L by inserting $L-1$ zeros between successive time samples. This creates $L-1$ images in the frequency domain, which are then removed by an L-band interpolated filter with spectral support from $-\pi/L$ to π/L. If L is sufficiently large, further interpolation with an irrational downsampling ratio can be achieved simply by a low-order interpolator such as Lagrange interpolation [29], cubic spline [30] and a low-order fractional-delay digital filter (FDDF) [28], etc. As an example, the cubic interpolator is able to provide rather accurate fractional delays up to about 0.1π. After which, both the amplitude and phase responses deviate considerably from an ideal FDDF [13], [31]. Therefore, an L-band interpolated filter should be used to upsample the input signal so that it can be fitted into the operating range of the cubic interpolator. It is also required to remove the images created by the upsampler due to the limited stopband attenuation of the cubic interpolator. One drawback of employing this structure in the SRR is that the output of the multistage decimators, which is obtained by downsampling the high-rate IF signal from the ADC, has to be upsampled again by the L-band filter. To overcome this problem, the functions of the L-band filter and the low-order interpolator can be simultaneously implemented using a VDF [6], [7], [32]. A VDF is a digital filter whose frequency and/or phase responses can be controlled by a parameter ϕ. The ideal frequency response of the VDF-based SRC is given by

$$H_{\mathrm{SRC}}^{(d)}(e^{j\omega}, \phi) = \begin{cases} e^{-j\omega\tau(\phi)}, & 0 \le |\omega| \le \omega_p \\ 0, & \omega_s \le |\omega| \le \pi \end{cases} \tag{2.20}$$

where $\tau(\phi) = D + \phi$ is the group delay of the SRC. ω_p and ω_s are the passband and stopband edges of the SRC, respectively. In the passband, it behaves like a FDDF with a parameter ϕ to provide the required arbitrary fractional delays. In the stopband, it helps to attenuate the undesirable frequency components. More precisely, the impulse response of the VDF, $h(n, \phi)$, is approximated by an L_{SRC}th-order polynomial in variable ϕ as follows:

$$h(n, \phi) = \sum_{l=0}^{L_{\mathrm{SRC}}-1} c_{l,n} \phi^l. \tag{2.21}$$

The z-transform of (2.21) is then given by

$$H_{\mathrm{SRC}}(z, \phi) = \sum_{l=0}^{L_{\mathrm{SRC}}-1} \left[\sum_{n=0}^{N-1} c_{l,n} z^{-n} \right] \phi^l = \sum_{l=0}^{L_{\mathrm{SRC}}-1} C_l(z) \phi^l \tag{2.22}$$

where $C_l(z) = \sum_{n=0}^{N-1} c_{l,n} z^{-n}$ are called the subfilters. (2.22) suggests a very useful structure, called the Farrow's structure [31], for implementing FDDFs and VDFs and it is shown in Fig. 10. It consists of a set of subfilters $C_l(z)$ followed by the multiplications with the appropriate powers of the parameter ϕ. It computes the required delayed (fractional) samples of the signal components in the passband, while attenuating those in the stopband. For modest downsampling ratios, the VDF-based SRC in Fig. 9(b) is more efficient than the structure in Fig. 9(a) because its coefficients can be jointly optimized to fulfill the given spectral and fractional-delay specifications. In the proposed SRR, the downsampling ratio of the SRC, M_{SRC}, is chosen to lie between 1 and 2. Thus, the VDF-based SRC leads to a better performance without having to increase the sampling rate as in the L-band filter approach [28]. As a result, the operating rate of the multistage decimators can be significantly lowered by a factor of L, say 4 to 8 in our example. In general, L will increase with the accuracy required.

The VDF-based SRC can be designed using the WLS [4], [6], [7] and SDP methods. For conciseness of presentation, only essential formulas are summarized as follows. Let $m = n + Nl$ and substituting $z = e^{j\omega}$ in (2.22), one gets

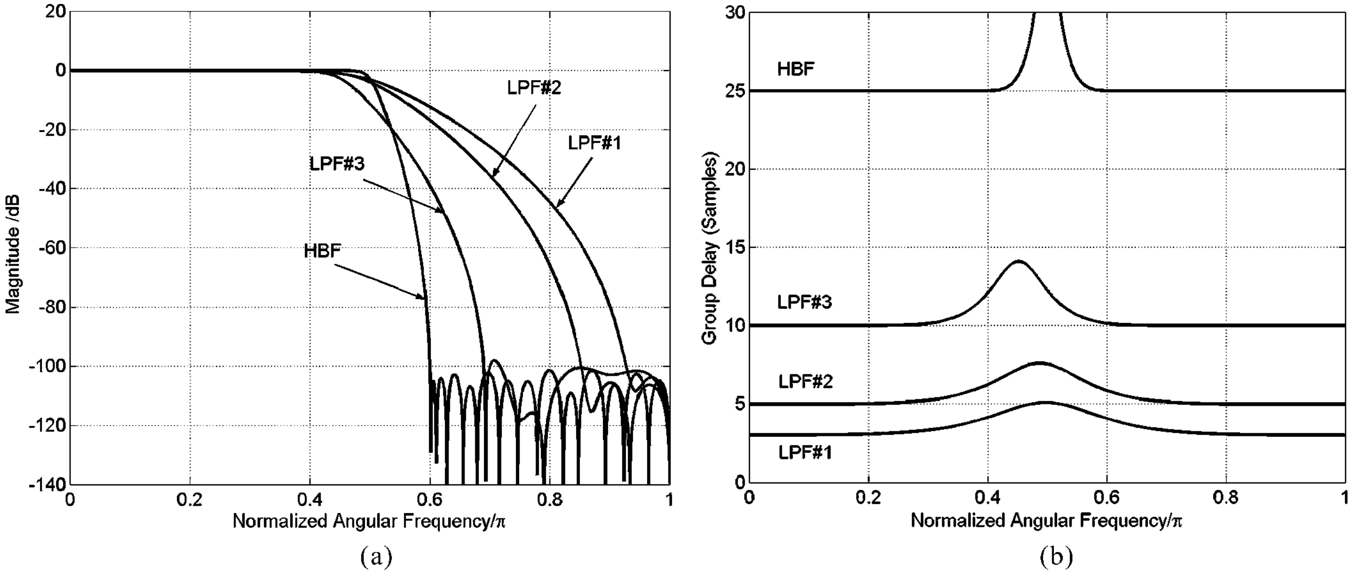
$$\begin{aligned} H_{\mathrm{SRC}}(e^{j\omega}, \phi) &= \sum_{l=0}^{L_{\mathrm{SRC}}-1} \left[\sum_{n=0}^{N-1} c_{l,n} e^{-j\omega n} \right] \phi^l \\ &= \sum_{m=0}^{L_{\mathrm{SRC}}N-1} a_m \beta_m(\omega, \phi) \\ &= \boldsymbol{a}^T \boldsymbol{\beta} \end{aligned} \tag{2.23}$$

where $\boldsymbol{a} = [a_0 \; a_1 \; \cdots \; a_{L_{\mathrm{SRC}}N-1}]^T$; $\boldsymbol{\beta} = [\beta_0(\omega, \phi), \ldots, \beta_{L_{\mathrm{SRC}}N-1}(\omega, \phi)]^T$; $a_m = c_{l,n}$; $\beta_m(\omega, \phi) = \phi^l e^{-j\omega n}$; $n = \mathrm{mod}(m, N)$ and $l = (m-n)/N$.

TABLE IV

SOPOT COEFFICIENTS OF ALLPASS FILTER IN LPF#3 (Filter Order $= 11, K_1 = 1, K_2 = 5$)

SOPOT Coefficients	SOPOT Coefficients
$b_{1,1} = 2^{-2} + 2^{-3} + 2^{-6} - 2^{-9} - 2^{-11} - 2^{-16}$	$b_{4,2} = 2^{-2} - 2^{-5} - 2^{-7} - 2^{-10} - 2^{-12} - 2^{-14} - 2^{-15}$
$b_{2,1} = -2^{-1} - 2^{-2} - 2^{-5} + 2^{-8} - 2^{-14} - 2^{-17}$	$b_{5,1} = 2^{-2} - 2^{-5} - 2^{-6} - 2^{-8} - 2^{-11} - 2^{-14} - 2^{-15} - 2^{-17}$
$b_{2,2} = 2^{-3} + 2^{-5} + 2^{-8} + 2^{-11} + 2^{-14} + 2^{-15} + 2^{-17}$	$b_{5,2} = 2^{-2} - 2^{-5} - 2^{-7} + 2^{-10} - 2^{-14} - 2^{-16}$
$b_{3,1} = 2^{-1} + 2^{-3} + 2^{-4} + 2^{-7} + 2^{-9} + 2^{-11} + 2^{-12} - 2^{-15} - 2^{-17}$	$b_{6,1} = -2^{-2} + 2^{-9} + 2^{-10} + 2^{-15}$
$b_{3,2} = 2^{-3} + 2^{-5} + 2^{-8} + 2^{-10} + 2^{-11} + 2^{-13} + 2^{-17}$	$b_{6,2} = 2^{-1} + 2^{-3} + 2^{-5} + 2^{-6} + 2^{-9} + 2^{-11} + 2^{-17}$
$b_{4,1} = -2^{-1} - 2^{-3} - 2^{-5} - 2^{-9} - 2^{-11} - 2^{-17}$	



(a) (b)

Fig. 7. (a) Frequency responses and (b) the corresponding group delays of the LPF#1, LPF#2, LPF#3, and HBF using allpass filters with fixed SOPOT coefficients.

1) Weighted Least Squares (WLS) Approach: In the WLS approach, the following least-squares cost function is minimized:

$$E_{LS} = \int_{\phi_s} \int_{\Omega_s} W(\omega, \phi) \cdot |E(\omega, \phi)|^2 \, d\omega d\phi \tag{2.24}$$

where $E(\omega, \phi) = H_{\mathrm{SRC}}(e^{j\omega}, \phi) - H_{\mathrm{SRC}}^{(d)}(e^{j\omega}, \phi)$; $W(\omega, \phi)$ is a positive weighting function; Ω_s is the spectral support over which $H_{\mathrm{SRC}}^{(d)}(e^{j\omega}, \phi)$ is to be approximated, and ϕ_s is the tuning space, which is chosen to be $\phi = [-0.5, 0.5]$. The solution can be obtained by solving the following linear equation:

$$\boldsymbol{Q}\boldsymbol{a}_{LS} = \boldsymbol{b} \text{ and } \boldsymbol{a}_{LS} = \boldsymbol{Q}^{-1}\boldsymbol{b} \tag{2.25}$$

where $[\boldsymbol{Q}]_{ij} = \int_{\phi_S} \int_{\Omega_S} W(\omega, \phi) \beta_i(\omega, \phi) \overline{\beta_j(\omega, \phi)} d\omega d\phi$; $[\boldsymbol{b}]_m = \int_{\phi_S} \int_{\Omega_S} W(\omega, \phi) \cdot \mathrm{Re}\{H_{\mathrm{SRC}}^{(d)}(e^{j\omega}, \phi) \overline{\beta_m(\omega, \phi)}\} d\omega d\phi$. It can be shown that the matrix $\boldsymbol{Q}$ is symmetric and positive definite. Conseuqently all its eigenvalues are distinct and real, and the matrix is nonsingular [20, pp. 55].

2) SDP Approach: The problem of designing the VDFs in the minimax sense can be formulated as

$$\min_{\boldsymbol{a}} \max_{-\pi < \omega < \pi} W(\omega, \phi) \left| H_{\mathrm{SRC}}(e^{j\omega}, \phi) - H_{\mathrm{SRC}}^{(d)}(e^{j\omega}, \phi) \right|. \tag{2.26}$$

Densely discretizing the frequency variable ω and the control parameter ϕ over the spaces Ω_s and ϕ_s into a set of points ω_{j_1}, $j_1 = 1, \ldots, J_1$, and ϕ_{j_2}, $j_2 = 1, \ldots, J_2$, we obtain the following equivalent problem of (2.26)

$$\min_{\boldsymbol{a}} \ \delta \text{ subject to } \delta - \left[\alpha_R^2(\omega_{j_1}, \phi_{j_2}) + \alpha_I^2(\omega_{j_1}, \phi_{j_2})\right] \geq 0, \tag{2.27}$$

for $j_1 = 1, \ldots, J_1$, and $j_2 = 1, \ldots, J_2$, where $\alpha_R(\omega, \phi) = W(\omega, \phi) |\mathrm{Re}\{H_{\mathrm{SRC}}(e^{j\omega}, \phi)\} - \mathrm{Re}\{H_{\mathrm{SRC}}^{(d)}(e^{j\omega}, \phi)\}|$; and $\alpha_I(\omega, \phi) = W(\omega, \phi) |\mathrm{Im}\{H_{\mathrm{SRC}}(e^{j\omega}, \phi)\} - \mathrm{Im}\{H_{\mathrm{SRC}}^{(d)}(e^{j\omega}, \phi)\}|$. Using Schur complement [10], the constraints in (2.27) can be rewritten in the following LMIs:

$$\boldsymbol{F}_{j_1, j_2}(\boldsymbol{a}) = \begin{bmatrix} \delta & \alpha_R(\omega_{j_1}, \phi_{j_2}) & \alpha_I(\omega_{j_1}, \phi_{j_2}) \\ \alpha_R(\omega_{j_1}, \phi_{j_2}) & 1 & 0 \\ \alpha_I(\omega_{j_1}, \phi_{j_2}) & 0 & 1 \end{bmatrix} \geq 0 \tag{2.28}$$

which is affine in the variable vector $\boldsymbol{a}$. Defining the augmented variable $\boldsymbol{x}^T = [\delta \ \boldsymbol{a}^T]$, the problem in (2.27) can be cast into the standard SDP problem in (2.17).

The WLS approach is attractive for its simplicity and fast design time. Additional linear equality constraints can also be incorporated using the Lagrange multiplier method, by solving

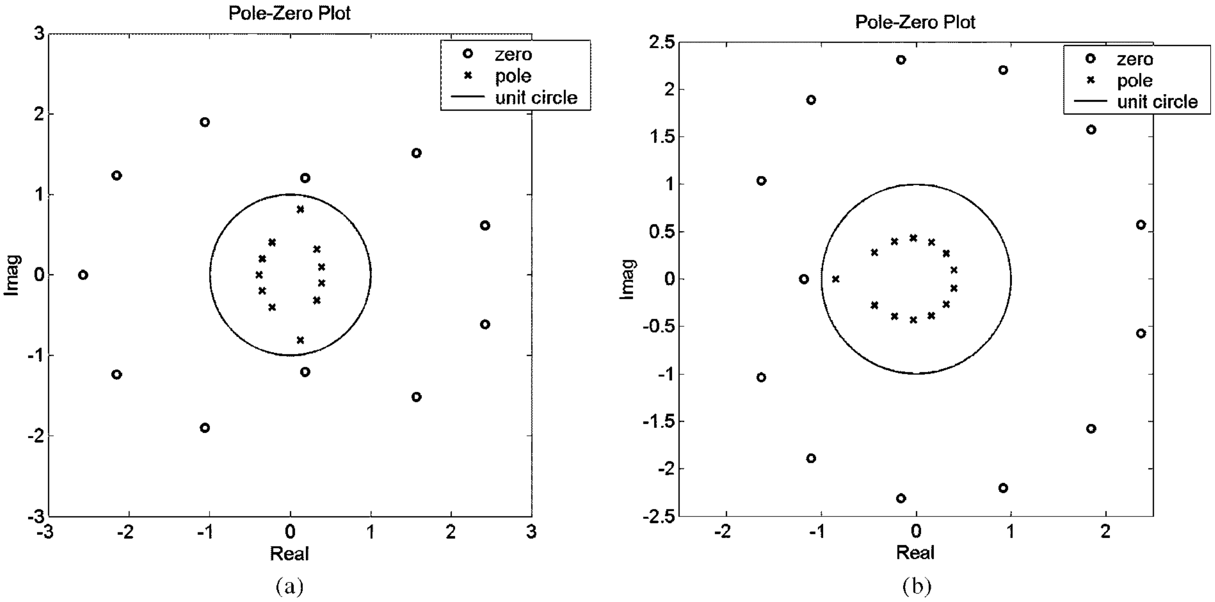


Fig. 8. Pole–zero plots of the digital allpass filters in a) LPF#3 (b) HBF.

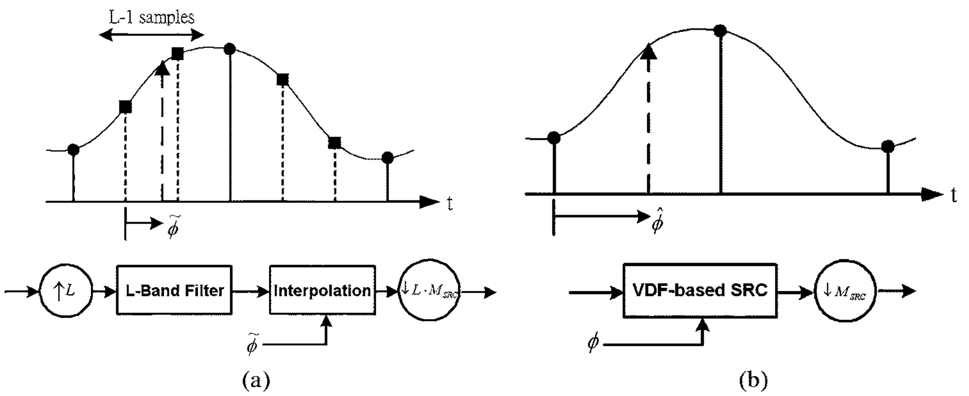


Fig. 9. SRC: (a) using L-band filter followed by a simple interpolation (Lagrange or cubic spline) and (b) using a VDF.

a quadratic programming problem with linearly equality constraints. It is also known as the eigenfilter method. For the VDF-based SRC, the design time for the WLS approach in a PIII-866 MHz personal computer is 14 s for $L_{\mathrm{SRC}} = 4$. For VDF with low order of interpolation, say $L_{\mathrm{SRC}} = 4$, the WLS and SDP approaches yield similar performance [32]. However, when L_{SRC} is increased and additional constraints are required, the SDP approach is more flexible and it yields better results, at the expense of more design time. For example, when L_{SRC} is increased to 7 with the same subfilter length, the worst-case stopband attenuation and the design time are respectively 93.2 dB and 28 s for the WLS approach, as compared to 98.1 dB and 43 min for the SDP approach (all the SDP designs are carried out using the LMI toolbox in MATLAB). Fig. 11 shows the corresponding frequency responses and the group delays of the VDF designed.

The multiplier-less realization of the VDF-based SRC was studied in [4]–[6]. In particular, all the subfilters in the Farrow structure are implemented in their transposed forms as shown in Fig. 10(b). By representing all these coefficients as SOPOT coefficients and employing the MB technique, the total number of additions can be kept to minimal by reusing the immediate results generated. As a result, the VDF-based SRC is free of variable multipliers except for the limited number of variable multipliers in the interpolation part of the Farrow structure. The specifications and the performances of the VDF-based SRC so obtained are summarized in Table I and their frequency responses are shown in Fig. 12. Due to page limitations, Table V only shows the SOPOT coefficients of the first subfilter $C_0(z)$. The results for other subfilters and the real-valued coefficients are omitted.

D. HBF

In this subsection, the design and implementation of the HBF shown in Fig. 2(b) is presented. In [4], the HBF is implemented using linear-phase FIR filters because it leads to more flexibility in choosing the cutoff frequency [4], [5]. Although HBFs has fewer nonzero coefficients than a general LPF, the difference in hardware complexity here is rather small when they are implemented as SOPOT coefficients and multiplier blocks. In order to reduce the system delay of the SRR, we propose to realize the

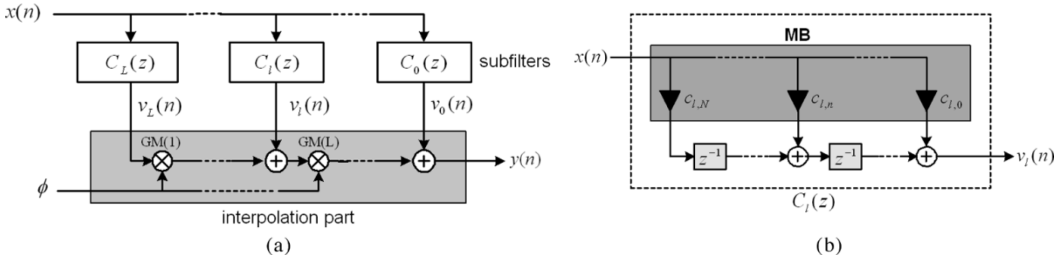


Fig. 10. Implementation of VDF-based SRC. (a) Farrow structure based. (b) Transposed form of FIR subfilters.

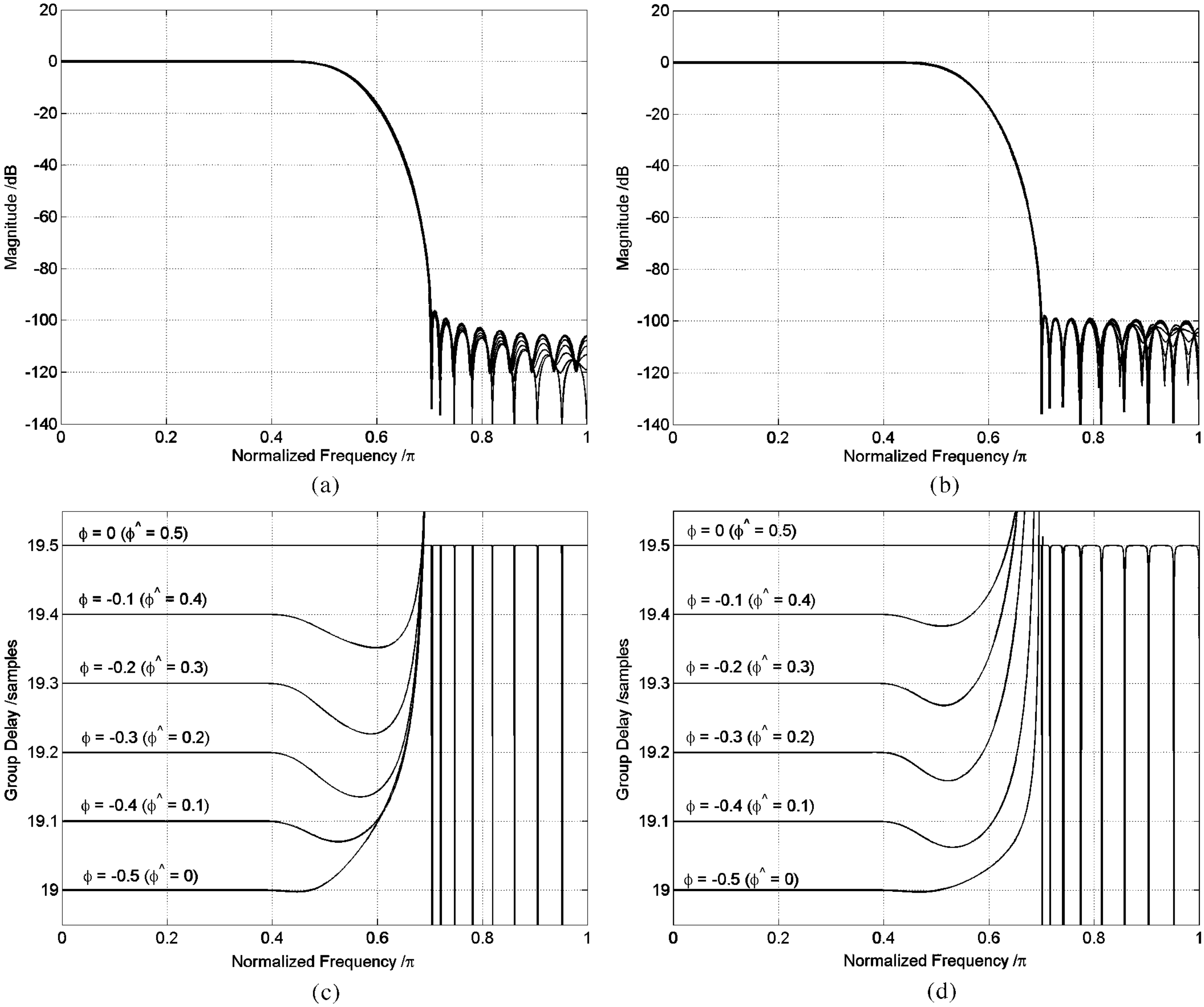


Fig. 11. (a) and (b) Frequency responses and (c) and (d) the corresponding group delays of the VDF-based SRC with real-valued coefficients and $\phi = \{-0.5, -0.4, -0.3, -0.2, -0.1, 0\}$ using WLS and SDP approaches, respectively. (subfilter length = 40, $L_{\mathrm{SRC}} = 7$).

HBF using low-delay FIRs and the allpass filters in (2.6) with the following transfer function:

$$H_{\mathrm{HBF}}(z) = \frac{\left[z^{-(2N-1)} + A_N(z^2)\right]}{2}. \qquad (2.29)$$

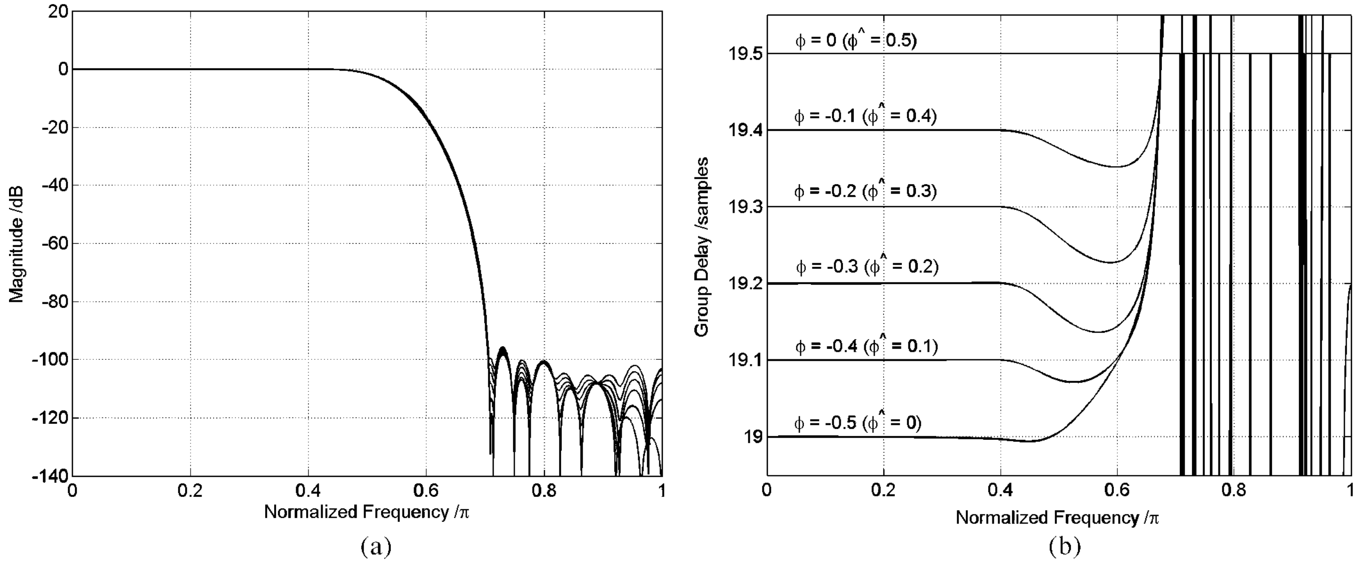
The desired phase response of the allpass filter is given by

$$\theta_{\mathrm{HBF}}^{(d)}(\omega) = -\left(N - \frac{1}{2}\right)\omega, \quad 0 < |\omega| < 2\omega_p \qquad (2.30)$$

where ω_p is the passband edge of the HBF. Note, due to the structural constraints of the HBF, the stopband edge ω_s is given by $\pi - \omega_p$. For the low-delay FIR HBF, $A_N(z^2)$ in (2.29) is replaced by a FIR function $F(z^2) = \sum_{n=0}^{L_{\mathrm{hbf}}-1} f_n \cdot z^{-2n}$ with the following desired response:

$$F_{\mathrm{HBF}}^{(d)}(e^{j\omega}) = e^{-j\left(N-\frac{1}{2}\right)\omega}, \qquad 0 < |\omega| < 2\omega_p \qquad (2.31)$$

where L_{hbf} is the filter length of the HBF. The design and multiplier-less realization of this HBF are also based on the methods described in Section II-B. Table I shows the specifications and the performances of the HBF, and its SOPOT coefficients are listed in Table VI. The frequency response and the group delays



(a)

(b)

Fig. 12. (a) Frequency responses and (b) the corresponding group delays of the VDF-based SRC with fixed SOPOT coefficients of FIR subfilters and $\phi = \{-0.5, -0.4, -0.3, -0.2, -0.1, 0\}$. (subfilter length $= 40, L_{\mathrm{SRC}} = 4$).

TABLE V
SOPOT Coefficients of FIR Subfilter $C_0(z)$ in VDF-Based SRC. (subfilter length $= 40, L_{\mathrm{SRC}} = 4$)

SOPOT Coefficients	SOPOT Coefficients
$c_{0,0} = c_{0,39} = 2^{-15} + 2^{-17}$	$c_{0,10} = c_{0,29} = -2^{-8} - 2^{-9} - 2^{-11} - 2^{-13} - 2^{-16} - 2^{-19}$
$c_{0,1} = c_{0,38} = 2^{-16} - 2^{-20}$	$c_{0,11} = c_{0,28} = 2^{-6} - 2^{-10} + 2^{-16}$
$c_{0,2} = c_{0,37} = -2^{-12} + 2^{-18}$	$c_{0,12} = c_{0,27} = 2^{-7} - 2^{-10} - 2^{-13} - 2^{-14} + 2^{-18} + 2^{-20}$
$c_{0,3} = c_{0,36} = 2^{-16} + 2^{-17} + 2^{-19}$	$c_{0,13} = c_{0,26} = -2^{-5} + 2^{-10} + 2^{-12} + 2^{-13} - 2^{-16} - 2^{-18} - 2^{-20}$
$c_{0,4} = c_{0,35} = 2^{-10} - 2^{-13} + 2^{-17} + 2^{-19}$	$c_{0,14} = c_{0,25} = -2^{-11} - 2^{-13} + 2^{-17} + 2^{-18}$
$c_{0,5} = c_{0,34} = -2^{-11} + 2^{-15} + 2^{-16} + 2^{-18}$	$c_{0,15} = c_{0,24} = 2^{-4} - 2^{-7} + 2^{-10} + 2^{-13} + 2^{-14} + 2^{-18} + 2^{-20}$
$c_{0,6} = c_{0,33} = -2^{-9} - 2^{-12} - 2^{-17} - 2^{-18}$	$c_{0,16} = c_{0,23} = -2^{-6} - 2^{-8} - 2^{-9} - 2^{-13}$
$c_{0,7} = c_{0,32} = 2^{-9} + 2^{-14} + 2^{-20}$	$c_{0,17} = c_{0,22} = -2^{-3} + 2^{-6} + 2^{-10} + 2^{-14} + 2^{-15}$
$c_{0,8} = c_{0,31} = 2^{-8} + 2^{-11} - 2^{-14} - 2^{-16}$	$c_{0,18} = c_{0,21} = 2^{-3} - 2^{-6} + 2^{-9} - 2^{-13} - 2^{-16} - 2^{-19} - 2^{-20}$
$c_{0,9} = c_{0,30} = -2^{-8} - 2^{-9} - 2^{-12} - 2^{-15} + 2^{-19}$	$c_{0,19} = c_{0,20} = 2^{-1} - 2^{-6} - 2^{-8} - 2^{-13} - 2^{-15} - 2^{-18} - 2^{-19}$

TABLE VI
SOPOT Coefficients of Allpass Filter in HBF (Filter Order $= 13, K_1 = 1, K_2 = 6$)

SOPOT Coefficients	SOPOT Coefficients
$b_{1,1} = -2^{-1} - 2^{-2} - 2^{-5} - 2^{-6} - 2^{-11} - 2^{-13} + 2^{-16}$	$b_{4,2} = 2^{-3} + 2^{-4} - 2^{-10} - 2^{-12} - 2^{-13} - 2^{-15} - 2^{-17}$
$b_{1,2} = 2^{-3} + 2^{-5} + 2^{-7} + 2^{-8} + 2^{-12} - 2^{-15} - 2^{-17}$	$b_{5,1} = 2^{-1} - 2^{-5} - 2^{-8} - 2^{-9} - 2^{-13} - 2^{-14} + 2^{-18}$
$b_{2,1} = -2^{-1} - 2^{-3} - 2^{-9} - 2^{-11} - 2^{-12} + 2^{-18}$	$b_{5,2} = 2^{-2} - 2^{-5} - 2^{-7} - 2^{-9} - 2^{-13}$
$b_{2,2} = 2^{-3} + 2^{-5} + 2^{-6} - 2^{-10} - 2^{-11} - 2^{-14} - 2^{-17}$	$b_{6,1} = 2^{0} - 2^{-3} - 2^{-10} - 2^{-11} - 2^{-14} - 2^{-16}$
$b_{3,1} = -2^{-2} - 2^{-4} - 2^{-7} - 2^{-9} - 2^{-11} - 2^{-13} + 2^{-16}$	$b_{6,2} = 2^{-2} + 2^{-6} + 2^{-9} + 2^{-14} + 2^{-16}$
$b_{3,2} = 2^{-3} + 2^{-5} + 2^{-6} + 2^{-8} - 2^{-12} - 2^{-15} + 2^{-18}$	$b_{7,1} = 2^{0} - 2^{-3} - 2^{-5} + 2^{-9} + 2^{-10} + 2^{-12} + 2^{-15} + 2^{-16}$
$b_{4,1} = 2^{-4} - 2^{-9} - 2^{-10} - 2^{-13} + 2^{-16}$	

of the allpass-based HBF are shown in Fig. 7. Fig. 8(b) shows its pole–zero plot.

III. Design Examples

In this section, we demonstrate the application of the proposed low-delay SRR to support the GSM, W-CDMA, CDMA2000, and Hiperlan/2 standards. The hardware complexities and the performances of the SRR using the allpass-based and low-delay FIR filters for both DSP and multiplier-less (hardware) implementations are examined and compared. A comparison between the proposed SRR and traditional programmable receivers is also presented. First of all, let us assume that the digitized IF signal is sampled at 80 M samples per

TABLE VII
CONFIGURATIONS AND COMPUTATIONAL COMPLEXITIES OF SRR FOR SUPPORTING GSM, W-CDMA, CDMA2000, AND HIPERLAN/2 STANDARDS. LD: LOW-DELAY. LP: LINEAR-PHASE. MOPS: MILLION OPERATIONS PER SECOND

Parameters	GSM	W-CDMA	CDMA2000	Hiperlan/2
Multiple Access	TDMA/FDMA	CDMA	CDMA	TDMA
Duplex Mode	TDD, FDD	FDD	FDD	TDD
Uplink Freq. Band (MHz)	890-915	1920-1980	Any existing band	5150-5350 5470-5725
Downlink Freq. Band (MHz)	935-960	2110-2170		
Channel Bandwidth (MHz)	0.2	5	1.25	20
Data Modulation	GMSK	QPSK	QPSK	OFDM
Input sampling rate (F_{s-in})	80 Msps			
Output sampling rate (F_{s-out})	270.833 ksps	3.84 Mcps	1.2288 Mcps	20 Msps
M_{SRR}	295.3849	20.83	65.1041	4
M_{CIC}	16	N/A	4	N/A
M_{SRC}	1.153847	1.302083	1.01725	N/A
k	4	4	4	2
Delay with LD (LP) (samples)	7146.31 (8012.77)	473.41 (533.49)	1675.8 (1870.56)	60 (71)
Proposed SRR using allpass-based (low-delay FIR) filters for DSP implementation				
Multiplications (MOPS)	107.42 (177.17)	1702.88 (2801.12)	434.32 (718.56)	1400 (3080)
Additions (MOPS)	565.71 (570.37)	3342.72 (3413.44)	1311.27 (1331.1)	2800 (2960)
Proposed SRR using allpass-based (low-delay FIR) filters for multiplier-less implementation				
Multiplications (MOPS)	1.625 (1.625)	23.04 (23.04)	7.3728 (7.3728)	0 (0)
Additions (MOPS)	864.99 (963.66)	8071.2 (9619.28)	2526.18 (2929.87)	9800 (10800)

TABLE VIII
PERFORMANCES OF PROPOSED SRR USING ALLPASS-BASED AND LOW-DELAY FIR FILTERS WITH REAL-VALUED AND SOPOT COEFFICIENTS FOR DIFFERENT OVERALL DOWNSAMPLING RATIOS, (PASSBAND DEVIATION, STOPBAND ATTENUATION, GROUP DELAY ERROR) IN DECIBELS

Range of M_{SRR}	Allpass (real-valued)	Allpass (SOPOT)	Low-delay FIR (real-valued)	Low-delay FIR (SOPOT)
$2 \le M_{SRR} \le 4$	(0.0096, 101.74, -50.17)	(0.0096, 100.48, -50.09)	(0.0096, 100.57, -49.27)	(0.0096, 100.03, -49.06)
$4 \le M_{SRR} \le 8$	(0.0090, 101.84, -44.75)	(0.0090, 100.53, -45.44)	(0.0089, 100.58, -44.72)	(0.0089, 100.03, -44.45)
$8 \le M_{SRR} \le 16$	(0.0082, 101.84, -40.07)	(0.0082, 100.58, -40.32)	(0.0082, 100.58, -39.31)	(0.0081, 100.03, -39.04)
$16 \le M_{SRR} \le 32$	(0.0063, 101.84, -36.07)	(0.0063, 100.62, -36.54)	(0.0064, 100.60, -35.46)	(0.0061, 100.14, -35.46)
$M_{SRR} \ge 32$	(0.0153, 101.77, -36.08)	(0.0147, 100.61, -36.56)	(0.0152, 100.58, -35.47)	(0.0146, 100.14, -35.46)

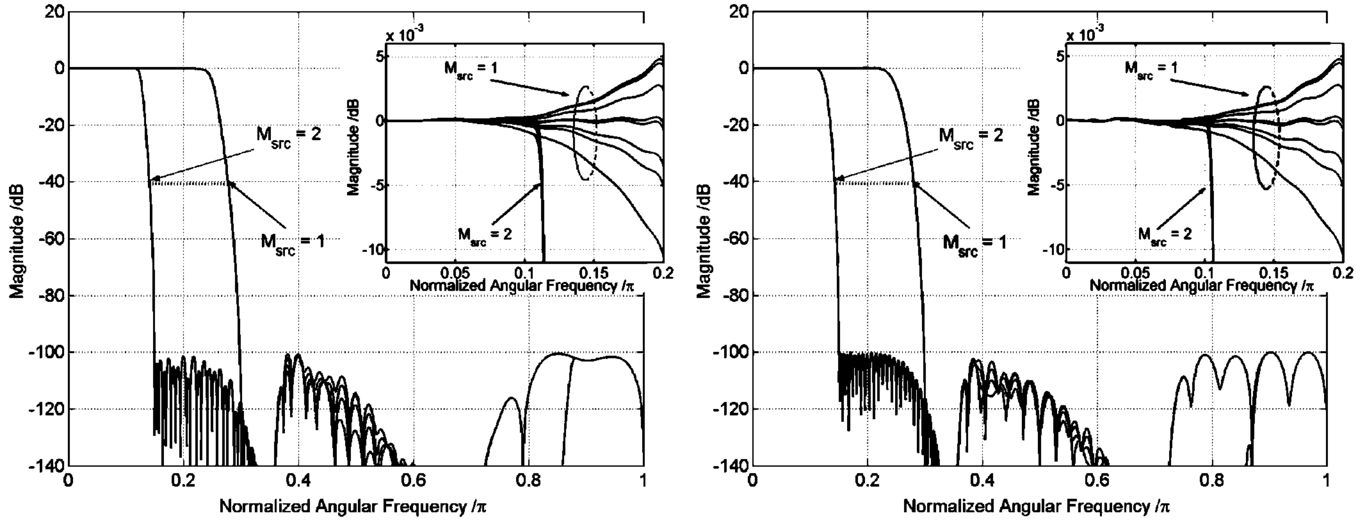
second (sps). Table VII summarizes some of the useful parameters for the GSM, W-CDMA, CDMA2000, and Hiperlan/2 standards [33]–[35]. It also includes the configurations and computational complexities for both DSP and multiplier-less implementations of the SRR. It can be seen that the computational complexities of the allpass-based SRR for the four communication standards are less than that using low-delay FIR filters for both DSP and multiplier-less implementations, especially for wideband application. In additional, the system delay of the low-delay SRR is (866.46, 60.08, 194.76, 11) samples lower (i.e., a reduction of 10.8%, 11.26%, 10.4% and 15%, respectively, in system delays) for GSM, W-CDMA, CDMA2000 and Hiperlan/2, respectively, as compared with their linear-phase counterpart, at the expense of modest increase in hardware complexity.

The target specifications of the SRR are 0.015 dB in passband deviation, 100 dB in stopband attenuation and -35 dB in fractional-delay error. By employing the random search algorithm [4], [13], the SOPOT coefficients of all the components, as shown in Tables I to VI, are obtained. Table VIII shows the passband deviations, stopband attenuations and group delay errors of the SRR using the allpass-based and low-delay FIR filters with both real-valued and SOPOT coefficients for different operating ranges of $M_{\rm SRR}$, i.e., cascading different components. It can be seen that the performances of the SRR using real-valued and SOPOT coefficients are similar. As an illustration, the frequency responses of the SRR with $4 \le M_{\rm SRR} \le 8$, i.e., cascading the LPF#3, HBF and the VDF-based SRC with $M_{\rm SRC} \in (1,2)$, using the SOPOT allpass-based and low-delay FIR filters are shown in Fig. 13. The frequency responses of the SRR, using the allpass-based filters with SOPOT coefficients, and the following operating ranges:

a) $2 \le M_{\rm SRR} \le 4$, i.e., cascading the HBF and the VDF-based SRC with $M_{\rm SRC} \in (1,2)$;

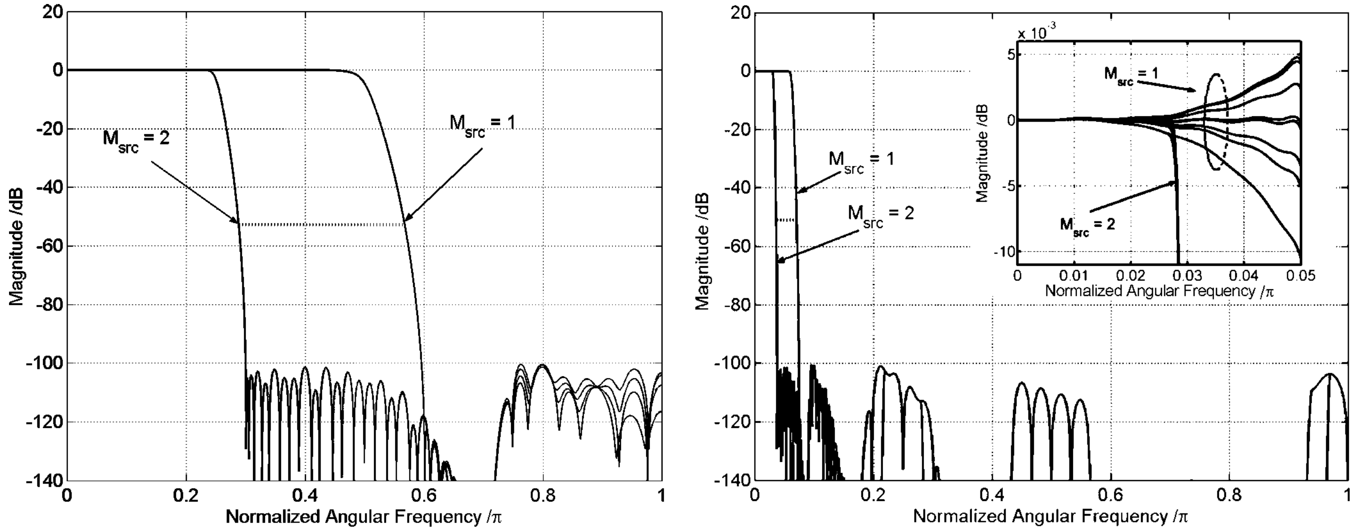
b) $16 \le M_{\rm SRR} \le 32$, i.e., cascading the LPF#1, LPF#2, LPF#3, HBF and the VDF-based SRC.

These are shown in Fig. 14(a) and (b), respectively. Since the proposed SRR is considerably different from the traditional programmable receiver, it is very difficult to make an exact comparison. In order to give the readers an idea of the potential benefits



(a) (b)

Fig. 13. Frequency responses of the SRR with $4 \leq M_{\rm SRR} \leq 8$, i.e., cascading the LPF#3, HBF and the VDF with $M_{\rm SRC} \in (1, 2)$, using (a) allpass-based (b) low-delay FIR filters with fixed SOPOT coefficients. (Note: the corresponding passband ripples are shown in the top right corner of the figures).



(a) (b)

Fig. 14. Frequency responses of the SRR with (a) $2 \leq M_{\rm SRR} \leq 4$, i.e., cascading the HBF and the VDF with $M_{\rm SRC} \in (1, 2)$ (b) $16 \leq M_{\rm SRR} \leq 32$, i.e., cascading LPFs, HBF, and VDF with $M_{\rm SRC} \in (1, 2)$ using allpass-based filters with fixed SOPOT coefficients.

and hardware savings of the proposed SRR, a comparison with the programmable receiver proposed in [2] is considered below. The architecture in [2] consists of a CIC filter with $L_{\rm CIC} = 4$, an ISOP sharpening filter, five modified HBFs (MHBFs) as the multistage decimators, and an PFIR. Since a SRC was not designed in [2], we assume that it is done using the same VDF-based SRC that we have proposed in Section II-C so that they have the same complexity. Furthermore, as the programmable receiver proposed in [2] is designed to be linear-phase, a SRR using the proposed technique but employing linear-phase FIR filters [4] is also included as a comparison.

Table IX shows the hardware complexities of the linear-phase SRR excluding the VDF-based SRC for the two receivers. It can be seen that the major hardware resources of the architecture in [2] is the variable multipliers required in the PFIR. Although the multiplications can be time multiplexed using a high-speed multiplier, it will limit the maximum clock speed of the receiver for wideband applications, i.e., small downsampling ratios. In the proposed SRR, the PFIR is replaced by a HBF with fixed coefficients, which results in very low implementation complexity, thanks to the novel VDF-based SRC. Therefore, the number of the variable multipliers can be drastically reduced. Note, the stopband attenuation of the linear-phase SRR is slightly lower than that in [2]. However, it considerably outperforms [2] in passband deviation and the number of variable multipliers as shown in Table IX. Table X shows the hardware complexities of the proposed low-delay SRR, excluding the VDF-based SRC using both allpass-based and low-delay FIR filters. It is observed that in order to reduce the system delay from (8012.77, 533.49, 1870.56, 71) to (7146.31, 473.41, 1675.8, 60) samples for GSM, W-CDMA, CDMA2000 and Hiperlan/2, respectively, the complexity is increased from 178 to 286 adders for the FIR realization. It can also be seen that for DSP implementation, the allpass-based SRR requires lower hardware cost (43 multipliers

TABLE IX
COMPARISON OF HARDWARE COMPLEXITIES BETWEEN [2] AND SRR USING LINEAR-PHASE FIR FILTERS FOR MULTIPLIER-LESS IMPLEMENTATION [4]. (NOTE: THE COMPLEXITIES OF VDF-BASED SRC ARE NOT INCLUDED)

	Architecture in [2]				SRR using linear-phase FIR			
Passband deviation	0.18 dB				0.015 dB			
Stopband attenuation	108 dB				100 dB			
	CIC	ISOP	MHBFs	PFIR	CIC	2nd order	LPFs	HBF
Multipliers	0	2	0	35	0	0	0	0
Adders	8	2	68	68	8	3	89	78
Total multipliers	37				0			
Total adders	146				178			

TABLE X
HARDWARE COMPLEXITIES OF PROPOSED SRR USING ALLPASS-BASED (LOW-DELAY FIR) FILTERS FOR DSP AND MULTIPLIER-LESS IMPLEMENTATIONS. (NOTE: COMPLEXITIES OF VDF-BASED SRC ARE NOT INCLUDED)

	DSP Implementation				Multiplier-less Implementation			
Passband deviation	0.015 dB							
Stopband attenuation	100 dB							
	CIC	2nd order	LPFs	HBF	CIC	2nd order	LPFs	HBF
Multipliers	0	0	21 (46)	13 (31)	0	0	0 (0)	0 (0)
Adders	8	3	42 (43)	26 (30)	8	3	137 (157)	87 (118)
Total multipliers	34 (77)				0 (0)			
Total adders	79 (84)				235 (286)			

TABLE XI
TOTAL NUMBER OF MULTIPLIERS AND ADDERS TO REALIZE PROPOSED SRR USING ALLPASS-BASED AND LOW-DELAY FIR FILTERS FOR DSP AND MULTIPLIER-LESS IMPLEMENTATIONS

	DSP Implementation		Multiplier-less Implementation	
	Allpass	Low-delay FIR	Allpass	Low-delay FIR
Multipliers	117	160	3	3
Adders	238	243	506	554

and five adders less) than that using the low-delay FIR filters. For multiplier-less (hardware) implementation, it still requires 51 fewer adders than the FIR realization. Table XI shows the total number of multipliers and adders required to implement the whole low-delay SRR. Note, though the multiplier-less SRR requires three variable multipliers in the interpolation part of the VDF-based SRC shown in Fig. 10(a), it is still much lower than the PFIR approach reported in [2]. Although not shown here due to page limitation, it is also possible to reduce the system delay in the traditional receiver by employing the techniques described in Section II-B. However, this receiver still requires considerable number of variable multipliers in the PFIR filter.

IV. OTHER IMPLEMENTATION ISSUES OF THE SRC

In this section, other implementation issues of the SRC for the SRR are presented. First of all, the flexible generation of the clocking signal for the sample rate conversion is very critical in order to support multiple standards in SRRs. In particular, it requires the generation of the clocking signals with different frequencies and high spectral purities. Although the sampling rate at the input of the SRR is fixed, the clock rate in the interpolation part of the VDF-based SRC has to be varied according to the required downsampling ratio of the receiver, which depends on the communication standard to be supported. These clocking signals can be generated by a direct digital frequency synthesizer (DDFS) [36]. The coordinate rotation digital computer (CORDIC)-based DDFS architecture proposed in [37] is particularly suitable for the SRC because of its high spectral purity and efficient multiplier-less realization. In the CORDIC-based DDFS, a digital sine wave with a certain frequency is generated by the CORDIC algorithm using a phase accumulator and a phase-to-amplitude converter. The DDFS is driven by a clock signal at a fixed frequency, which is considerably higher than the frequencies to be generated. At each time instant, the appropriate values of the sine wave is calculated using the CORDIC algorithm, which can be performed by a sequence of shift-and-add operations. This yields an efficient multiplier-less implementation of the DDFS with high phase resolutions, high precision, and low spur-free dynamic range (SFDR). To generate the required clocking signal, the digital values of the reference sine wave generated by the DDFS is sent to a digital-to-analog converter (DAC) to produce a staircase-like analog approximation of the sine wave. After appropriate low-pass filtering, a comparator can be used to generate the desired binary clocking signal.

Another implementation issue of the SRC is the calculation of the fractional-delay parameter ϕ for each output sample of

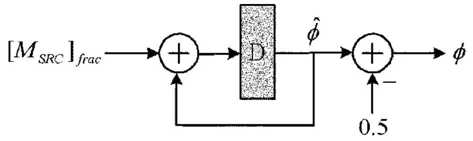


Fig. 15. Implementation of the fractional-delay calculation unit.

the SRC. This requires a fractional-delay calculation unit shown in Fig. 15, which is based on the fractional part of M_{SRC}, to calculate the required ϕ value for each output sample of the SRC. For example, as shown in Fig. 12(b), if $M_{\mathrm{SRC}} = 1.2$, the fractional delay $\hat{\phi}_1$ is 0.2 for the first output sample of the SRC. The corresponding fractional-delay parameter ϕ_1 is equal to -0.3. Similarly, the second output sample has a fractional delay $\hat{\phi}_2$ of 0.4 and ϕ_2 is equal to -0.1, and so on. In general, let ϕ_k be the fractional-delay parameter at the kth output sample of the SRC. The fractional-delay parameter ϕ_k can be computed from the fractional part of M_{SRC}, Fig. 15, as follows:

$$\phi_k = [k \cdot [M_{\mathrm{SRC}}]_{\mathrm{frac}}]_{\mathrm{frac}} - 0.5 \qquad (4\text{-}1)$$

where $k = 1, 2, \ldots$ and $[\alpha]_{\mathrm{frac}}$ denotes the fractional part of the value α.

V. Conclusion

The design and multiplier-less realization of a new SRR with reduced system delays is presented. Its employs low-delay FIR and digital allpass filters to effectively reduce the system delay of the multistage decimators in SRRs. The optimal least-square and minimiax designs of these low-delay FIR and allpass-based filters are formulated as a SDP problem, which allows zero magnitude constraint at $\omega = \pi$ to be incorporated readily as additional LMIs. By implementing the sampling rate conversion using a VDF immediately after the integer decimators, the needs for an expensive programmable FIR filter in the traditional SRR is avoided. The design of the VDF-based SRC using the WLS and SDP methods are formulated and compared. Other implementation issues including the multiplier-less and DSP realizations of the various digital filters and the generation of the clock signal in the SRC are also studied. Design results show that the proposed architecture considerably reduces the system delay and implementation complexities (especially in high-speed variable multipliers) as compared with conventional approaches.

References

[1] T. Hentschel and G. Fettweis, "Sample rate conversion for software radio," *IEEE Commun. Mag.*, pp. 142–150, Aug. 2000.

[2] H. J. Oh, S. Kim, G. Choi, and Y. H. Lee, "On the use of interpolated second-order polynomials for efficient filter design in programmable downconversion," *IEEE J. Select. Areas Commun.*, pp. 551–560, Apr. 1999.

[3] A. Y. Kwentus, Z. Jiang, and A. N. Willson, "Application of filter sharpening to cascaded integrator-comb decimation filters," *IEEE Trans. Signal Processing*, vol. 45, pp. 457–467, Feb. 1997.

[4] K. S. Yeung and S. C. Chan, "On the design and multiplier-less realization of digital IF for software radio receivers," in *Proc. EUSIPCO*, vol. 1, Sept. 2002, pp. 695–698.

[5] S. C. Chan and K. S. Yeung, "On the design and multiplier-less realization of digital IF for software radio receivers with prescribed output accuracy," in *Proc. DSP*, vol. 1, 2002, pp. 277–280.

[6] C. K. S. Pun, S. C. Chan, K. S. Yeung, and K. L. Ho, "On the design and implementation of FIR and IIR digital filters with variable frequency characteristics," *IEEE Trans. Circuits Syst. II*, vol. 49, pp. 689–703, Nov. 2002.

[7] T. B. Deng, "Weighted least-squares method for designing arbitrarily variable 1-D FIR digital filters," in *Proc. Signal Processing*, 2000, pp. 597–613.

[8] Y. C. Lim and S. R. Parker, "FIR filter design over a discrete power-of-two coefficient space," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 583–591, Apr. 1983.

[9] C. K. S. Pun and S. C. Chan, "The minimax design of digital all-pass filters with prescribed pole radius constraint using semidefinite programming (SDP)," in *Proc. ICASSP*, vol. 6, Apr. 2003, pp. 413–416.

[10] H. Wolkowicz, R. Saigal, and L. Vandenberghe, *Handbook of Semidefinite Programming—Theory, Algorithms, and Applications*. Boston, MA: Kluwer, 2000.

[11] W. S. Lu, "Design of nonlinear-phase FIR digital filters: a semidefinite programming approach," in *Proc. ISCAS*, vol. 3, 1999, pp. 263–266.

[12] C. Y. Fung and S. C. Chan, "A multistage filterbank-based channelizer for software radio base stations," in *Proc. IEEE ISCAS*, vol. 3, 2002, pp. 429–432.

[13] C. K. S. Pun, Y. C. Wu, S. C. Chan, and K. L. Ho, "An efficient design of fractional-delay digital FIR filter using Farrow structure," in *Proc. 11th IEEE Signal Processing Workshop on Statistical Signal Processing*, 2001, pp. 595–598.

[14] A. G. Dempster and M. D. MacLeod, "Use of minimum-adder multiplier blocks in FIR digital filters," *IEEE Trans. Circuits Syst. II*, pp. 569–577, Sept. 1995.

[15] S. K. Mitra, *Digital Signal Processing: A Computer-Based Approach*. Singapore: McGraw-Hill, 1998.

[16] T. Saramaki, Y. Neuvo, and S. K. Mitra, "Design of computationally efficient interpolated FIR filters," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 70–88, Jan. 1988.

[17] G. Wade, A. Roberts, and G. Williams, "Multiplier-less FIR filter design using a genetic algorithm," *IEEE Vision, Image and Signal Processing*, vol. 141, pp. 175–180, June 1994.

[18] J. Yli-Kaakinen and T. Saramaki, "An efficient algorithm for the design of lattice wave digital filters with short coefficient wordlength," in *Proc. ISCAS*, vol. 3, 1999, pp. 443–448.

[19] C. C. Chen and A. N. Willson, "A trellis search algorithm for the design of FIR filters with signed-powers-of-two coefficients," *IEEE Trans. Circuits Syst. II*, vol. 46, pp. 29–39, Jan. 1999.

[20] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[21] X. Zhang and H. Iwakura, "Design of IIR digital allpass filters based on eigenvalue problem," *IEEE Trans. Signal Processing*, vol. 47, pp. 554–559, Feb. 1999.

[22] M. Lang, "Allpass filter design and applications," *IEEE Trans. Signal Processing*, vol. 46, pp. 2505–2514, Sept. 1998.

[23] T. Q. Nguyen, T. I. Laakso, and R. D. Koilpillai, "Eigenfilter approach for the design of allpass filters approximating a given phase response," *IEEE Trans. Signal Processing*, vol. 42, pp. 2257–2263, Sept. 1994.

[24] T. Saramaki and M. Renfors, "A remez-type algorithm for designing digital filters composed of all-pass sections based on phase approximations," in *Proc. 38th Midwest Symp. Circuits and Systems*, vol. 1, Aug. 1995, pp. 571–575.

[25] K. Steiglitz and J. F. Kaiser, "METEOR: a constraint-based FIR filter design program," *IEEE Trans. Signal Processing*, vol. 40, pp. 1901–1909, Aug. 1992.

[26] A. G. Dempster and M. D. Macleod, "Comparison of IIR filter structure complexities using multiplier blocks," in *Proc. ISCAS*, vol. 2, May 1995, pp. 858–861.

[27] T. A. Ramstad, "Digital methods for conversion between arbitrary sampling frequencies," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 577–591, June 1984.

[28] T. Saramaki and T. Ritoniemi, "An efficient approach for conversion between arbitrary sampling frequencies," in *Proc. ISCAS*, vol. 2, 1996, pp. 285–288.

[29] G. S. Liu and C. H. Wei, "Programmable fractional sample delay filter with Lagrange interpolation," *Electron. Lett.*, vol. 26, no. 19, pp. 1608–1610, 1990.

[30] J. M. de Carvalho and J. V. Hanson, "Efficient sample rate conversion with cubic splines," in *Proc. Telecommunications Symp.*, 1990, pp. 439–442.

[31] C. W. Farrow, "A continuously variable digital delay element," in *IEEE Int. Symp. Circuits and Systems*, 1988, pp. 2641–2645.

[32] S. C. Chan and K. S. Yeung, "On the application of variable digital filters (VDF) to the realization of software radio receivers," in *Proc. ISCAS*, vol. 3, Bangkok, Thailand, May 2003, pp. 562–565.

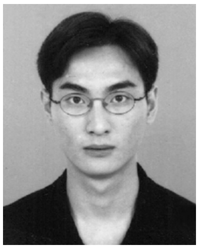
[33] A. Mehrotra, *Cellular Radio: Analog and Digital Systems*. Boston, MA: Artech House, 1994.

[34] T. Ojanpera and R. Prasad, *WCDMA: Toward IP Mobility and Mobile Internet*. Boston, MA: Artech House, 2001.

[35] ETSI HIPERLAN/2 Standard [Online]. Available: http://www.etsi.org/technicalactiv/Hiperlan/hiperlan2.htm

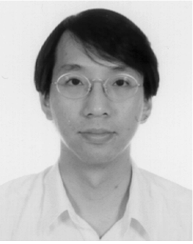
[36] A. Madisetti, A. Y. Kwentus, and A. N. Willson, "A 100-Mhz, 16-b, direct digital frequency synthesizer with a 100-dBc spurious-free dynamic range," *IEEE J. Solid-State Circuits*, vol. 34, pp. 1034–1043, Aug. 1999.

[37] F. C. Tormo and J. V. Coquillat, "Optimization of direct digital frequency synthesizers based on CORDIC," *Electron. Lett.*, vol. 37, no. 21, pp. 1278–1280, 2001.



K. S. Yeung received the B.Eng. degree in electrical and electronic engineering from The University of Hong Kong, Hong Kong, in 2001. He is currently pursuing M.Phil. degree at the Department of Electrical and Electronic Engineeringat the same university.

His main research interests are in digital signal processing, multirate filter-banks and wavelets, digital filter design, and their efficient realization and applications.



S. C. Chan (S'87–M'92) received the B.Sc. (Eng.) and Ph.D. degrees from the University of Hong Kong, Hong Kong, in 1986 and 1992, respectively.

He joined the City Polytechnic of Hong Kong, Hong Kong, in 1990 as an Assistant Lecturer and later as a University Lecturer. Since 1994, he has been with the Department of Electrical and Electronic Engineering, the University of Hong Kong, and is now an Associate Professor. He was a Visiting Researcher with Microsoft Corporation, Redmond, CA, and Microsoft, Beijing, China, in 1998 and 1999, respectively. His research interests include fast transform algorithms, filter design and realization, multirate signal processing, communications signal processing, and image-based rendering.

Dr. Chan is currently a member of the Digital Signal Processing Technical Committee of the IEEE Circuits and Systems Society. He was Chairman of the IEEE Hong Kong Chapter of Signal Processing from 2000 to 2002.