

Signal Processor Implementation of Digital All-Pass Filters

MARKKU RENFORS, MEMBER, IEEE, AND EVANGELOS ZIGOURIS

Abstract—Digital filter structures composed of all-pass subfilters have turned out to be very useful in various applications. In this paper, we investigate the implementation of such filters using modern high-speed signal processors. We concentrate on digital all-pass filters realized as a cascade of first- and second-order all-pass sections. First, we consider wave digital all-pass sections with special emphasis on developing two-port adaptor structures which are suitable for signal processor implementation. New explicit formulas are developed for scaling and round-off noise analysis of these all-pass sections. We also consider other well-known first- and second-order all-pass filter sections, including direct-form structures and the structures developed by Mitra and Hirano. The performances of the different all-pass filter structures are compared with respect to roundoff noise, parasitic oscillations, and speed of signal processor implementation.

I. INTRODUCTION

IN recent years, a large number of digital filter structures have been developed. The main concern has been the performance of the digital filter in finite wordlength implementation, on one hand, and the computational complexity of the implementation, on the other. It has turned out that very useful digital filter structures for various applications can be constructed by using all-pass subfilters as building blocks. Traditionally, all-pass filters have been used as *phase equalizers* [1]. Also, *conventional digital filters* (e.g., classical odd-order low-pass and high-pass transfer functions) can be realized as a parallel connection of two all-pass filters. A well-known class of such filters are the lattice wave digital filters [2], [3] which are related to certain analog prototype networks. Fig. 1 illustrates a certain type of lattice WDF where the all-pass filters are realized as a cascade of first- and second-order all-pass sections [4]. It should be emphasized that these filters can be designed directly from an appropriate z -domain transfer function without referring to the analog prototype network [5], [6]. Direct z -domain techniques have also been advanced for designing certain extended filter types [7].

As a special case of lattice WDF's, half-band IIR filters have been found to be very useful in those *sampling rate conversion* applications where the sampling rate alteration factor is two (or power of two in multistage realizations) [4], [5], [8]–[12]. Recently, recursive N th-band filters

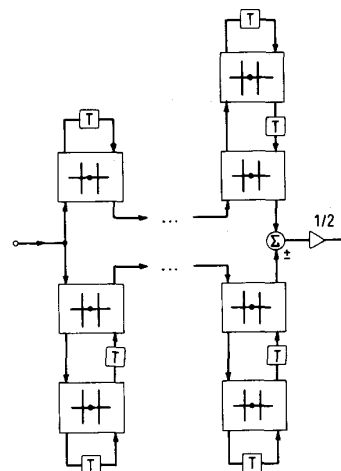


Fig. 1. A lattice wave digital filter.

[10], [11], [13]–[15] were introduced for more general sampling rate conversion applications. These filters are realized as a parallel connection of N all-pass filters where N is the decimation or interpolation ratio.

Lattice WDF's, and also recursive N th-band filters, are well suited to *filter bank applications*. Since a pair of complementary filters can be implemented with the cost of a single filter, lattice WDF's are very efficient in tree-type filter banks [12], [16], [17]. Furthermore, uniform filter banks can be realized efficiently using recursive N th-band filters and discrete Fourier transform [13], [18]–[20].

All the filter classes mentioned above can also be designed to have *approximately linear phase response* in the passband [11], [14], [21], [22]. This is achieved by selecting one of the all-pass branch filters to contain only delay elements. In the passband region, the phase responses of the all-pass branches must be approximately equal because the output signals of the parallel all-pass branches should be in phase. Therefore, the other all-pass branches, and consequently the overall filter, are forced to approximate closely the linear phase response of the delay branch in the passband. Such approximately linear phase designs are especially suitable for applications where linear phase FIR filters would have excessive signal delay.

The different classes of filters composed of all-pass subfilters have turned out to be very effective in their re-

Manuscript received September 25, 1986; revised October 16, 1987. This work was supported in part by the Academy of Finland.

M. Renfors is with the Department of Electrical Engineering, Tampere University of Technology, P.O. Box 527, SF-33101 Tampere, Finland.

E. Zigouris is with the Electronics Laboratory, Department of Physics, University of Patras, 261 10 Patras, Greece.

IEEE Log Number 8719430.

0096-3518/88/0500-0714\$01.00 © 1988 IEEE

spective applications, for instance, in terms of multiplication rate and number of delay elements. In addition, all these filter types can be realized by using first- and second-order all-pass sections as basic building blocks. The resulting filter structures are highly modular, which makes them suitable for signal processor implementations.

Recently, all-pass filters with complex coefficients have also received considerable attention [23]. It has been shown that using a parallel connection of two all-pass filters with complex conjugate coefficients, e.g., even-order classical low-pass designs can be realized. Even though the complex all-pass filter-based structures share many of the important properties of the corresponding real all-pass filter-based structures, they are quite different from the implementation point of view. Therefore, we concentrate here on real all-pass filters.

In this paper, we discuss the properties of various real first- and second-order all-pass sections. The main goal is to find the structures which are most suitable for implementation with modern one-chip signal processors. Due to the limited wordlengths of current signal processors, it is necessary to pay close attention to the finite wordlength properties (coefficient sensitivity, roundoff noise, and parasitic oscillations) of the filter structures used in the implementation. Designing the all-pass filters to have good finite wordlength properties results in good finite wordlength performance of the overall filter in all the filter classes mentioned above.

An important, but often overlooked, aspect in digital filter implementation is parasitic oscillation. Especially, it is important that the filter has stable forced response under overflow nonlinearities [24], [25]. Even though conservative scaling criterion could be used to eliminate all arithmetic overflows, it does not guarantee that the filter recovers from overflow situations caused by bit faults which may occur in prolonged operation independently of the input signal. Therefore, forced response stability is a necessary requirement in many practical applications where the filter is expected to operate over long periods of time. In this respect, wave digital filters have the desirable property that stable forced response can be guaranteed by using saturation arithmetic [24], [25]. Using appropriate all-pass filter structures, this property can be generalized to the other filter classes mentioned above.

In Section II of this paper, we shall discuss the essential properties of modern high-speed signal processors, such as NEC μ PD7720 [26] and Texas Instruments TMS320 family [27]. In Section III, we concentrate on implementing all-pass filters using the circulator structure, i.e., as a cascade of first- and second-order WDF all-pass sections [4], [28] containing two-port adaptors (see Fig. 1). New explicit formulas for scaling and roundoff noise analysis are developed. In Section IV, we discuss certain other all-pass filter structures including direct-form sections and Mitra-Hirano structures [29]. In Section V, the properties of the all-pass filter structures are compared. Finally, in Section VI, we give an example of a practical signal processor implementation of a lattice WDF.

II. DIGITAL SIGNAL PROCESSORS

Programmable single-chip digital signal processors are becoming increasingly popular for implementing high-performance signal processing systems. In this paper, we consider two different architectures, NEC μ PD7720 [26] and Texas Instruments TMS320 family [27]. In the latter case, we concentrate on the processor version TMS32010, but most of the conclusions apply also to the later versions TMS32020 and TMS32025. It should be emphasized that the purpose is not to compare the efficiency of these two processor architectures, but to examine the implications of modern signal processor architecture to the design of digital filter structures.

The characteristics of these two processors are summarized in Table I. Both processors have highly pipelined architectures utilizing hardware multipliers which give the results in one instruction cycle. The TMS32010 has a single 32-bit accumulator, while the μ PD7720 has two 16-bit accumulators which can be used, at lower speed, as a single 32-bit accumulator. Due to special pipelined architectures, both of these processors are especially suitable for inner product type of computations, i.e., for computing expressions of the form

$$\sum_{i=1}^N a_i x_i$$

where x_i and a_i are data values and (predefined) coefficients, respectively. In the case of TMS32010, once the inner product computation has been initiated, each term takes two instruction cycles and the result is obtained with 32-bit accuracy. In the case of μ PD7720, once the computation has been initiated, each term of the inner product takes one (two) cycles with 16-bit (32-bit) accumulation.

In order to achieve smooth overflow behavior and stable forced response in recursive digital filters, saturation arithmetic is often utilized [24], [25]. When computing an inner product, it is desirable that the saturation characteristic is applied to the overall result and not during the intermediate computations. Both processors considered here have special facilities for implementing saturation arithmetic. In the case of μ PD7720, saturation arithmetic can be implemented easily in inner products containing up to four terms (three additions). This processor has a special flag indicating overflows. Testing the flag and loading the correct saturation value from a special register takes two or one instruction cycles depending on whether an overflow has occurred or not. In the case of TMS32010, the processor can be set to a mode where it makes the saturation correction automatically after every addition. Using this saturation mode, the desired type of saturation characteristic is obtained only in certain special cases, i.e., when an overflow may occur only in the last addition of the inner product.

The computational accuracy of both the TMS32010 and μ PD7720 is basically 16 bits, even though 13-bit coefficient wordlength gives certain advantages because such coefficients can be stored outside the RAM memory. The

TABLE I
DIGITAL SIGNAL PROCESSOR FEATURE SUMMARY

Feature	NEC μ PD7720	TI TMS32010
Data wordlength (bits)	16	16
Coefficient wordlength (bits) (ROM / RAM)	13 / 16	13 / 16
Accumulator width (bits)	16	32
Multiplier precision (bits) (in $\times$ in = out)	$16 \times 16 = 31$	$16 \times 16 = 32$
Instruction cycle (ns)	250	200
Data RAM (bits)	128×16	144×16
Coefficient ROM (bits)	512×13	*
Program memory (words)		
On chip	512	1536
Full-speed external	—	4096
Parallel I/O (bits)	8	16

* 13-bit coefficients can be stored in the program memory.

16-bit data wordlength means that, in the case of recursive digital filters, the algorithm has to be designed carefully in order to keep the roundoff noise level and parasitic oscillations sufficiently low. In many applications, it is impossible to get sufficient dynamic range using the conventional cascade of direct-form sections. It is also well known that the problem becomes more difficult for narrow-band (or extremely wide-band) low-pass and high-pass filters.

Traditionally, the computational complexity of digital filter structures has been evaluated in terms of the number of multipliers. However, when modern digital signal processors are used for the implementation, the number of multipliers is not a very critical factor. It is more important that the structure is regular, preferably based on inner product building blocks.

III. WAVE DIGITAL ALL-PASS FILTER STRUCTURES

In this section, we consider first- and second-order all-pass filter sections similar to those used in the specific lattice WDF structure of Fig. 1. As seen from the figure, such all-pass filters are composed of unit delays and two-port adaptors. The important properties of the all-pass filter implementation, such as speed and scaling of internal signal levels, are determined mainly by the two-port adaptor structures used. In this section, we first discuss two-port adaptor structures from the signal processor implementation point of view. Then explicit formulas are developed for scaling and roundoff noise analysis of the first- and second-order WDF all-pass filters.

A. Adaptor Structures

A two-port adaptor is defined by the equations [31]

$$b_1(n) = -\gamma a_1(n) + (1 + S\gamma) a_2(n) \quad (1a)$$

$$b_2(n) = (1 - S\gamma) a_1(n) + \gamma a_2(n) \quad (1b)$$

where

$$S = \pm 1. \quad (1c)$$

In (1), a_i and b_i , $i = 1, 2$, are the input and output signals, respectively, of port i . For a stable filter, the adaptor coef-

ficient γ satisfies

$$-1 < \gamma < 1. \quad (2)$$

As will be discussed later in this section, the sign-parameter S has an effect on the scaling of internal signal levels.¹

Fig. 2 shows three different two-port adaptor structures which all satisfy (1). The structure of Fig. 2(a) is one of the traditional two-port adaptor structures [31] containing only one multiplier. The structures of Fig. 2(b) and (c) contain a higher number of multipliers than necessary but, as will be seen in Section V-D, they are practically as fast to implement using modern signal processors as the traditional structures. In addition, these structures have certain advantages with respect to scaling of signal levels and implementation of saturation arithmetic.

In order to preserve the good stability properties of WDF's under finite precision arithmetic, it is essential that the quantization and overflow nonlinearities are located at the adaptor outputs and not at the internal computations of the adaptors² [32]. Since we want stable forced response to be guaranteed, it is especially important that overflows do not occur inside the adaptors, and that overflow detection and saturation corrections are performed at the adaptor outputs. To achieve this in the traditional two-port adaptor structures, such as Fig. 2(a), it is necessary to use a guard-bit in the internal computations. Especially, a guard-bit is needed after the summation of the adaptor input signals, i.e., at the multiplier input. In a signal processor implementation, the use of a guard-bit reduces the number of fractional bits by one and hence reduces the dynamic range by 6 dB. In the adaptor structures of Fig. 2(b) and (c), additions are performed only at the output nodes, and we can use the special features of the μ PD7720 and TMS32010 processors for implementing saturation arithmetic without the need for a guard-bit.

We use L_2 and L_∞ norms of the scaling transfer functions for estimating the internal signal levels of the filter implementation [33]. In digital filter structures composed of first- and second-order all-pass sections, the signal levels in each section can be scaled independently of the others; this is because the all-pass factors of the scaling transfer functions do not have any effect on the scaling norms. The scaling is basically accomplished by properly selecting the sign parameters S in the two-port adaptors [4]. In many cases, this results in acceptable scaling. However, depending on the location of the pole realized by the filter section, additional scaling coefficients are often needed.

The conventional way of scaling wave digital filters, based on inserting ideal transformers in the analog prototype network [31], is illustrated in Fig. 3(a). This kind of scaling has drawbacks: some additional operations are

¹In case of traditional two-port adaptor structures, changing the sign parameter is equivalent to transposing the structure [31].

²These nonlinearities can also be located inside the adaptors if the quantization and overflow characteristics are selected appropriately. However, we feel that this is not practical in signal processor implementations.

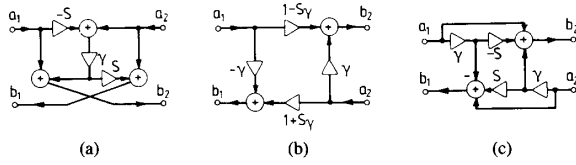
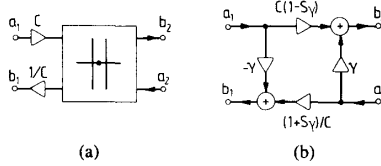
Fig. 2. Two-port adaptor structures. $S = \pm 1$.

Fig. 3. Scaling of two-port adaptors.

needed, and it is not possible to optimally scale all relevant nodes in the network. However, in the four-multiplier adaptor structure, the scaling coefficients can be incorporated into the adaptor coefficients [34]; the arrangements of Fig. 3(a) and (b) are equivalent if the internal computations are executed accurately.

Because the frequency response of the lattice filter structure of Fig. 1 is rather sensitive to gain errors in the branch filters, it is important that the filter structure realizes an all-pass transfer function also with quantized coefficient values. This means that in Fig. 3(a), C must be a (positive or negative) power of 2 so that both scaling coefficients have an accurate finite-length binary representation. In the case of Fig. 3(b), there are more choices for the scaling coefficient which helps in optimizing the dynamic range. Assuming that the adaptor coefficient is first quantized to finite wordlength, the basic criterion is that the coefficients

$$\begin{aligned} c_1 &= (1 - S\gamma_q)C \\ c_2 &= (1 + S\gamma_q)/C \end{aligned} \quad (3)$$

must have finite-length binary representations.

Let us define the integer coefficients

$$\begin{aligned} C_1 &= 2^b(1 - S\gamma_q) \\ C_2 &= 2^b(1 + S\gamma_q) \end{aligned} \quad (4)$$

where γ_q is the adaptor coefficient quantized to $1 + b$ bit wordlength. Then the acceptable values of the scaling coefficient are of the form

$$C = \frac{2^n f_2}{f_1} \quad (5)$$

where f_1 and f_2 are arbitrary integer factors of C_1 and C_2 , respectively. In other words, the scaling coefficient is a power of 2 multiplied by a fractional number such that the denominator factors are cancelled out when computing c_1 and c_2 according to (3).

The optimal solution is the one which gives a scaling norm closest to (but less than) unity and gives acceptable wordlengths for c_1 and c_2 . In our case, with the μ PD7720 and TMS32010 signal processors, the fractional part should be 12 or 15 bits depending on how the coefficients are stored. Obviously, the acceptable values of the scaling coefficient depend on the quantized value of the adaptor coefficient γ_q . Therefore, in order to obtain finer tuning of the signal levels, it is sometimes worth considering different values of γ_q in the neighborhood of the minimum-length solution.

B. First-Order All-Pass Section

A first-order all-pass filter section realizing the transfer function

$$A_1(z) = \frac{z^{-1} - \gamma}{1 - \gamma z^{-1}} \quad (6)$$

is shown in Fig. 4. For general analysis, the scaled four-multiplier adaptor is used.

The output signal of this structure satisfies the L_p scaling criterion automatically for any value of p . Thus, there is only one node to be scaled, the input of the delay element. The scaling transfer function from input to this node is given by

$$F(z) = \frac{C(1 - S\gamma)}{1 - \gamma z^{-1}} \quad (7)$$

and the corresponding L_2 and L_∞ norms are [35]

$$\|F\|_2^2 = C^2 \frac{1 - S\gamma}{1 + S\gamma} \quad (8a)$$

$$\|F\|_\infty = C \frac{1 - S\gamma}{1 - |\gamma|}. \quad (8b)$$

We can see that by selecting $C = 1$ and $S = \text{sign}[\gamma]$, we get an L_∞ -scaled realization [4], i.e., $\|F\|_\infty = 1$. Notice also that the adaptor structure of Fig. 2(c) with $S = \text{sign}[\gamma]$ results in an L_∞ -scaled first-order all-pass section.

Even if the adaptor structure of Fig. 2(a) is used, the signal level at the delay element is scaled as above. In this case, the maximum signal level inside the adaptor (at the input of the multiplier) may be twice as high as at the adaptor inputs. Overflows in the internal computations are avoided by using a guard-bit and performing saturation arithmetic at the output nodes.

In the roundoff noise analysis of the structure of Fig. 4, we use the standard uncorrelated noise model [33]. Both rounding after multiplications and rounding after accumulations (at the adaptor outputs) are considered. The noise transfer functions from the quantization points to the output are given by

$$G_1(z) = 1 \quad (9a)$$

$$G_2(z) = \frac{(1 + S\gamma)z^{-1}}{C(1 - \gamma z^{-1})}. \quad (9b)$$

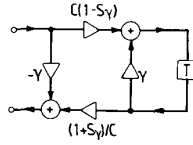


Fig. 4. First-order WDF all-pass structure.

Consequently, the output roundoff noise power is given by

$$\sigma_1^2 = \nu \left(1 + \frac{1 + S\gamma}{C^2(1 - S\gamma)} \right) \sigma_0^2, \quad (10a)$$

where σ_0^2 is the noise power of a unit noise source and

$$\nu = \begin{cases} 1 & \text{for rounding after accumulations} \\ 2 & \text{for rounding after multiplications.} \end{cases} \quad (10b)$$

If the scaling coefficient $C = 1$, we actually use the two-port adaptor structures of Fig. 2. Even in the case of Fig. 2(b), the noise sources are correlated and the previous derivation of the noise model is not necessarily valid. However, in Appendix A, it is shown that using proper noise models for these structures, the same result (10a), with $C = 1$, is obtained for the output noise power. The only exception is that for the structure of Fig. 2(a), $\nu = 1$ in (10a) also with rounding after multiplications. However, due to the need for a guard-bit, the structure of Fig. 2(a) results in lower dynamic range than the other adaptor structures even in this case.

Experience has shown that the above uncorrelated noise model gives a good estimate for the output quantization noise power, as well as for the noise spectrum, if quantization is performed by (two's-complement) rounding. With two's-complement truncation, the actual noise spectrum is close to the estimated one except for a dc offset. In the case of magnitude truncation, the noise power is usually higher, and a good estimate can be obtained only by simulation.

C. Second-Order All-Pass Section

The second-order all-pass filter section shown in Fig. 5 realizes the transfer function

$$A_2(z) = \frac{-\gamma_1 + \gamma_2(\gamma_1 - 1)z^{-1} + z^{-2}}{1 + \gamma_2(\gamma_1 - 1)z^{-1} - \gamma_1 z^{-2}}. \quad (11)$$

Also, in this structure, the adaptor output signal levels are to be scaled. The L_p scaling norms at the two branches between the adaptors are equal. This follows from the fact that the two scaling transfer functions differ only by an all-pass factor. Therefore, it is sufficient to scale the signal levels at two nodes, the inputs of the delay elements. As shown in Appendix B, the L_2 and L_∞ scaling norms are given by

$$\|F_1\|_2^2 = C_1^2 \frac{1 - S_1\gamma_1}{1 + S_1\gamma_1} \quad (12a)$$

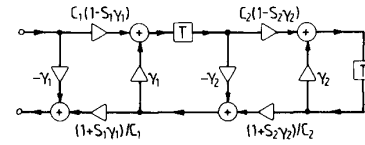


Fig. 5. Second-order WDF all-pass structure.

$$\|F_2\|_2^2 = C_1^2 C_2^2 \frac{(1 - S_1\gamma_1)(1 - S_2\gamma_2)}{(1 + S_1\gamma_1)(1 + S_2\gamma_2)} \quad (12b)$$

$$\|F_1\|_\infty = C_1 \frac{1 - S_1\gamma_1}{1 - |\gamma_1|} \quad (13a)$$

$$\|F_2\|_\infty = C_1 C_2 (1 - S_1\gamma_1)(1 - S_2\gamma_2) f \quad (13b)$$

where

$$f = \begin{cases} \frac{1}{(1 - \gamma_1)(1 - |\gamma_2|)} & \text{when } |\gamma_2| \geq \frac{-4\gamma_1}{(1 - \gamma_1)^2} \\ \sqrt{\frac{4\gamma_1}{(1 + \gamma_1)^2(4\gamma_1 + \gamma_2^2(1 - \gamma_1)^2)}} & \text{otherwise.} \end{cases} \quad (13c)$$

By selecting $C_1 = 1$ and $S_1 = \text{sign}[\gamma_1]$, we obtain $\|F_1\|_\infty = 1$ and the signal levels in the two branches between the adaptors are optimally L_∞ scaled. If we then select $C_2 = 1$ and $S_2 = \text{sign}[\gamma_2]$, we obtain $\|F_2\|_\infty \leq 1$. For example, for low-pass lattice WDF's with moderate bandwidth, this scaling norm is typically close to unity, and satisfactory scaling is obtained just by selecting the sign parameters properly. If the bandwidth is very narrow or very wide, this scaling norm is often much less than unity and a proper value for C_2 must be selected to obtain acceptable scaling.

The output roundoff noise power, using the same assumption as in the case of first-order all-pass filters, is given by (see Appendix B)

$$\sigma_2^2 = \nu \left(1 + 2 \frac{1 + S_1\gamma_1}{C_1^2(1 - S_1\gamma_1)} + \frac{(1 + S_1\gamma_1)(1 + S_2\gamma_2)}{C_1^2 C_2^2 (1 - S_1\gamma_1)(1 - S_2\gamma_2)} \right) \sigma_0^2 \quad (14)$$

where ν is given by (10b).

An interesting point to notice here is that if we select the scaling multipliers in (8a) and (12) in such a way that the L_2 scaling norms become equal to unity,³ the contribution of each noise source is σ_0^2 . Consequently, the total output roundoff noise power for a cascade of first- and

³With such scaling, the structures become equivalent to first- and second-order all-pass four-multiplier Gray-Markel lattices [36].

second-order all-pass sections becomes $2\nu N\sigma_0^2$, where N is the order of the all-pass filter.

IV. OTHER ALL-PASS FILTER STRUCTURES

A. Direct-Form Structures

We consider here the first- and second-order direct-form 1 structures [1] shown in Fig. 6(a) and (b), respectively. The transfer functions of these networks are given by

$$A_1(z) = \frac{b_1 + z^{-1}}{1 + b_1 z^{-1}} \quad (15)$$

$$A_2(z) = \frac{b_2 + b_1 z^{-1} + z^{-2}}{1 + b_1 z^{-1} + b_2 z^{-2}}. \quad (16)$$

All signal levels in these structures satisfy automatically both L_2 and L_∞ scaling criteria. The noise transfer functions are just the all-pole transfer functions corresponding to (15) and (16). Hence, the noise gains are

$$\sigma_1^2 = \nu \frac{1}{1 - b_1^2} \quad (17)$$

$$\sigma_2^2 = \nu^2 \frac{1 + b_2}{(1 - b_2)((1 + b_2)^2 - b_1^2)} \quad (18)$$

where ν is given by (10b).

One drawback of the direct-form 1 sections is that the structures are not canonic with respect to the number of delays. However, in a cascade of all-pass sections, the delay elements of two consecutive sections can be combined [29], as shown in Fig. 6(c). Therefore, only two extra delays are required independently of the order of the all-pass filter. Also, the number of multipliers is higher than necessary, but otherwise the structures are very regular containing only one inner product.

In direct-form 2 sections [1], the roles of the scaling and noise transfer functions are reversed comparing to the direct-form 1 structures. Therefore, additional scaling coefficients are needed at the input and output of the filter sections. The noise gains of L_2 -scaled filters are the same as above (ignoring the effect of additional quantization points after the scaling coefficients) but the noise gains of L_∞ -scaled filters may be considerably higher.

B. Mitra-Hirano Structures

A large variety of first- and second-order all-pass structures were presented by Mitra and Hirano in [29]. It is interesting to note that the structures 1B and 1B_i of Mitra and Hirano are equivalent to the first-order WDF all-pass structures based on the adaptor of Fig. 2(a) with $S = +1$ and $S = -1$, respectively. The structures 1A_i and 3D can be obtained from the direct-form 1 sections by combining the multipliers with equal coefficients.

In most cases, additional scaling coefficients are needed at the input and output of these structures in order to scale the internal signal levels. Analytical expressions for the L_2 scaling norms can be found from [6], and the uncorrelated noise formulas for the unscaled filters are given in

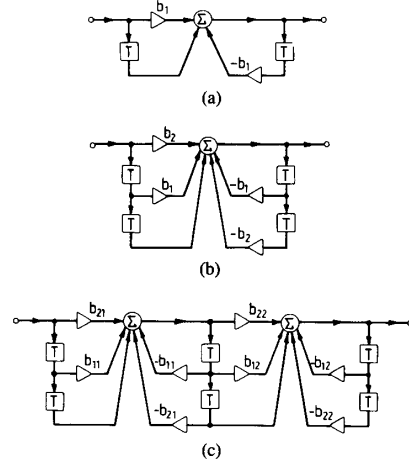


Fig. 6. Direct-form 1 all-pass structures. (a) First order. (b) Second order. (c) Delay sharing.

[29]. We have used numerical techniques to compute also the L_∞ scaling norms.

V. COMPARISONS

A. Roundoff Noise

Here we compare the roundoff noise performance of the all-pass filter structures described above. Figs. 7 and 8 show the noise gains versus pole locations for the first- and second-order (complex pole) sections, respectively, using both L_2 and L_∞ scaling. The scaling coefficients, when required, were selected to satisfy the scaling criterion accurately. In practice, suboptimal scaling, e.g., with integer or power of two coefficients is often preferred and the noise gains are slightly higher. The structures included in the comparison are:

- Dm* Direct-form 1 structure with rounding after multiplications ($\nu = 2$)
- Da* Direct-form 1 structure with rounding after accumulations ($\nu = 1$)
- Mm* Best Mitra-Hirano structure for each pole location with rounding after multiplications
- Ma* Best Mitra-Hirano structure for each pole location with rounding after accumulations
- Wm* WDF structure with rounding after multiplications ($\nu = 2$)
- Wa* WDF structure with rounding after accumulations ($\nu = 1$).

For the Mitra-Hirano structures, only the lowest noise gain for each pole location is shown. The additional noise sources after the scaling coefficients were not taken into consideration since their effect on the overall noise is small. The differences between the roundoff noise properties of the first-order Mitra-Hirano structures are not very significant (at most 3 dB difference in noise gain), but in the second-order case the differences are much larger. It is interesting to know which of the Mitra-Hirano

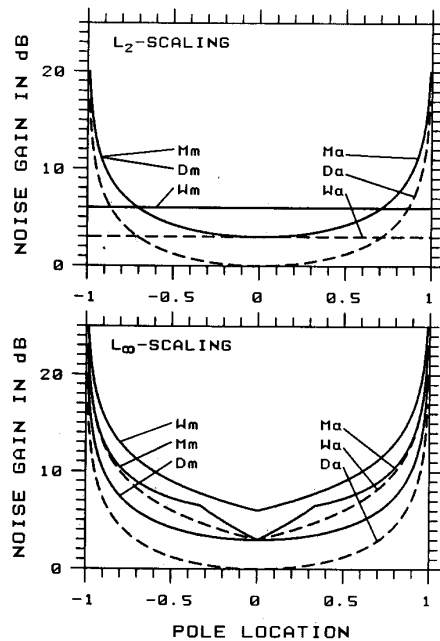


Fig. 7. Noise gain versus pole location for different first-order all-pass structures.

structures gives the lowest noise gains for high values of pole radius r since these sections are decisive for the total output noise of a filter. Using L_∞ scaling with $r > 0.9$, the structure $3D$ gave the best results for all-pole angles. Using L_2 scaling with $r > 0.9$ and rounding after accumulations (after multiplications), the structures $3C$, $3D$, and $3B$ ($3C_i$, $3D_i$, and $3B_i$) gave the best results for pole angles close to 0 , $\pi/2$, and π , respectively.

From Fig. 8, we can see that if the pole realized by a second-order all-pass filter is close to the point $z = 1$ or $z = -1$, then the WDF structure gives significantly lower noise gains than any other structure included in the comparison. This is true also in the first-order case with L_2 scaling. In other cases, especially when the pole is close to the imaginary axis, the direct-form 1 sections give very good results.

A comparison of the roundoff noise properties of different all-pass filter types in actual filter realizations can be seen in Fig. 9. It shows the overall noise gains for fifth-order elliptic low-pass filters versus passband edge. The filters have 0.1 dB passband ripple and 60 dB stopband attenuation. It can be seen that in narrow-band and extremely wide-band cases, the WDF all-pass structures give the lowest noise gains. With moderate bandwidths, the direct-form 1 sections give the best results except for the case of L_2 scaling with rounding after multiplications, where the Mitra-Hirano structures are the best ones with respect to roundoff noise.

B. Coefficient Sensitivity

In general, filters which are realized as a parallel connection of two or more all-pass subfilters are very sensitive to gain errors in the all-pass branch filters. Therefore,

it is important to use all-pass filter structures which realize all-pass transfer functions also with quantized coefficient values. This condition is a prerequisite for the property of structural boundedness and, consequently, for the low passband sensitivity of these kinds of filters [6]. This condition is satisfied in all structures considered in this paper provided that the coefficients are quantized appropriately. For example, in the four-multiplier adaptor structure [Figs. 2(b) and 3(b)], it is necessary that the parameter γ is first quantized and the actual filter coefficients are computed accurately using this quantized value.

A first-order all-pass section is specified by a single parameter which is the same for all structures considered in this paper. Consequently, also the coefficient sensitivity properties of different first-order sections are equivalent. In the second-order structures, three different parameter sets have been used: 1) WDF structures [cf. (11)]; 2) direct-form and Mitra-Hirano types 3 [cf. (16)]; 3) Mitra-Hirano types 2 (cf. [29]).

For a lattice WDF, the coefficient quantization effects are usually insignificant in the passband region. To compare the sensitivities of different second-order sections, we have determined the minimum coefficient wordlength for a number of low-pass filters such that the reduction in stopband attenuation is less than 1 dB comparing to the original design with high-precision coefficients. In all cases considered, the passband ripple remained below 0.12 dB (originally 0.1 dB). The results are shown in Fig. 10. No systematic differences were observed between the three parameterizations of the transfer function. It can also be seen that the required coefficient wordlength increases with increasing stopband attenuation. The usual 16-bit wordlength is sufficient in the fifth-order case with 60 dB stopband attenuation [see Fig. 10(a)] except for very narrow-band or wide-band filters. Also, in the case of seventh-order filters with 80-dB stopband attenuation [Fig. 10(b)], the 16-bit wordlength is sufficient for moderate bandwidths if a somewhat larger margin is allowed for coefficient quantization or if some kinds of optimized quantization techniques are applied.

C. Parasitic Oscillations

Our main concern here are the overflow oscillations. Especially, we consider the problem how to achieve stable forced response under overflow nonlinearities. This means that the filter is guaranteed to recover from an overflow even with nonzero input signals. In general, it has been shown [24] that a filter has stable forced response if the following two conditions are satisfied:

- i) the filter has no zero-input overflow oscillations with a two's-complement type of overflow characteristic; and
- ii) suitable overflow characteristic (e.g., saturation arithmetic) is used.

In wave digital filters, stable forced response can be achieved by using saturation arithmetic at the adaptor outputs [24], [30]. Also, in the other types of first-order all-pass structures, stable forced response can be achieved by using saturation arithmetic at appropriate locations in the

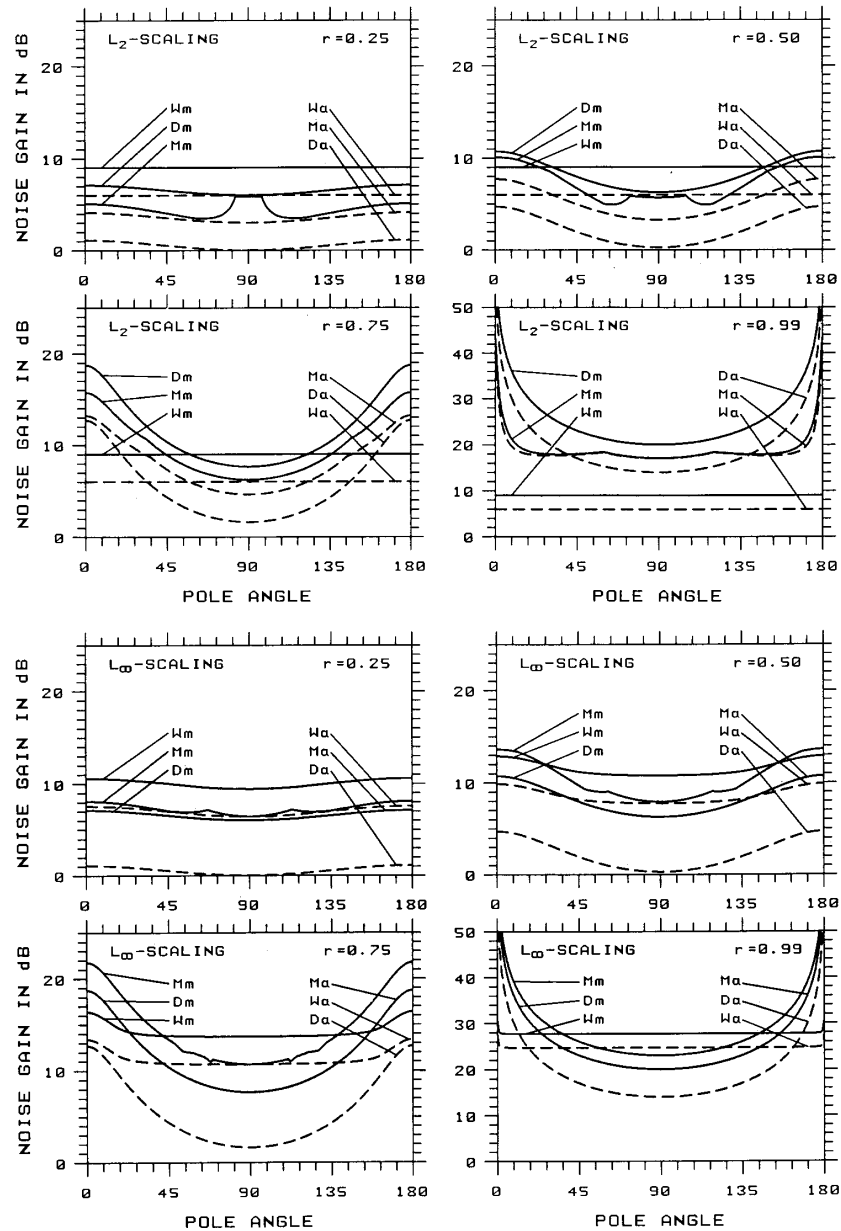


Fig. 8. Noise gain versus pole location for different second-order all-pass structures.

feedback loops. However, in the second-order case, stable forced response can be guaranteed for direct-form 1 sections and for Mitra-Hirano all-pass sections only with certain restricted pole locations.

It is well known [37], [38] that a second-order direct-form IIR filter has no zero-input overflow oscillations when the denominator coefficients [cf. (16)] satisfy

$$|b_1| + |b_2| < 1. \quad (19)$$

Therefore, in second-order direct-form all-pass structures, a stable forced response can be achieved using saturation arithmetic after the summation node if the coefficients of the all-pass section satisfy (19). The same is true also for Mitra-Hirano-type 3A, 3B, 3C, and 3D networks whose recursive part has the same structure. Equation (19) is a very restrictive constraint. In the low-pass example of Section V-A (0.1 dB passband ripple, 60 dB stopband attenuation), this condition is satisfied in all sections only

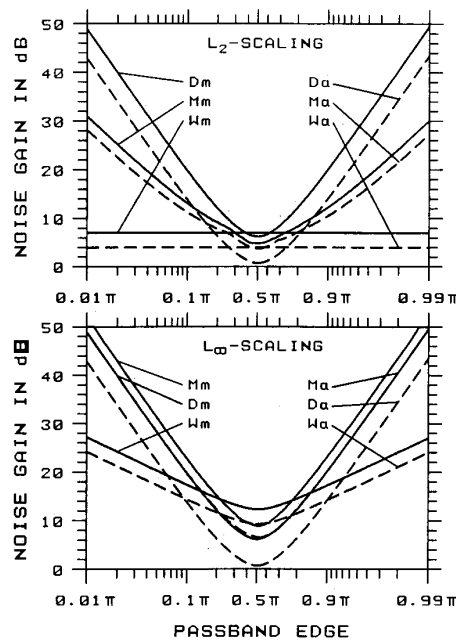


Fig. 9. Noise gain versus passband edge for elliptic low-pass filters with 0.1 dB passband ripple and 60 dB stopband attenuation.

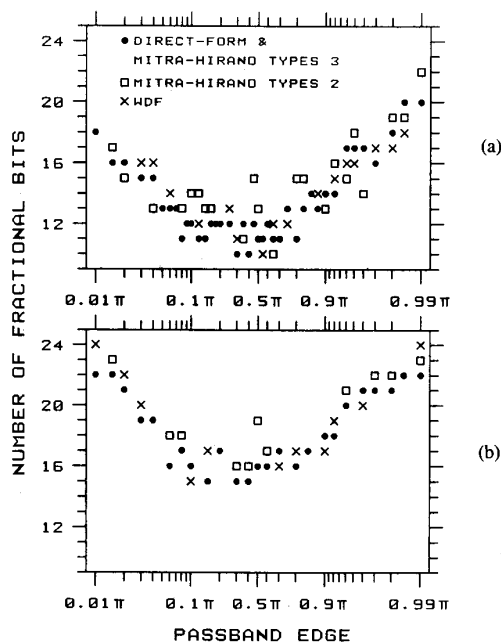


Fig. 10. Coefficient wordlengths for elliptic low-pass filters required to maintain the stopband attenuation within 1 dB from the original design. (a) Filter order = 5, passband ripple = 0.1 dB, stopband attenuation = 60 dB. (b) Filter order = 7, passband ripple = 0.1 dB, stopband attenuation = 80 dB.

when the passband edge is in the range $0.434\pi < \omega_p < 0.515\pi$.

In the WDF structures, also zero-input quantization limit cycles can be eliminated by using magnitude truncation at the adaptor outputs [32]. In the other second-

order all-pass structures, these limit cycles can be eliminated only for certain restricted pole locations even if magnitude truncation is used. On the other hand, in most applications, it is not adequate to eliminate only zero-input limit cycles since parasitic oscillations may still appear with constant nonzero input signals. In addition, magnitude truncation, which is usually required to achieve limit-cycle-free realizations, tends to increase somewhat the roundoff noise power comparing to rounding. Furthermore, the implementation of magnitude truncation would require some extra instructions, at least in the two signal processors considered in this paper. For these reasons, we do not consider the use of magnitude truncation for eliminating parasitic oscillations. An interesting topic would be to compare the limit-cycle amplitudes appearing in different second-order all-pass structures using simulation or by developing bounds for the maximum amplitudes [30], but this is outside the scope of this paper.

D. Signal Processor Implementation

Tables II and III compare the execution times of certain first- and second-order all-pass structures implemented using both μ PD7720 and TMS32010/32020 signal processors. The tables show the execution times for the subroutines (μ PD7720) or macros (TMS320) implementing the all-pass sections, not including the implementation of saturation arithmetic. The length of the main program, i.e., the number of instructions needed for composing the overall filter from the all-pass sections, is rather independent of the all-pass subfilter structure. Therefore, Tables II and III can be used as a basis for comparing different structures. They do not give the right picture about the relative efficiency of the two processors in this application because the number of additional instructions required in practical implementations is quite different for the two processors. In addition, the implementations based on the two signal processors are not quite comparable with respect to finite wordlength effects. The μ PD7720 implementations utilize 16-bit accumulation, corresponding to two's-complement truncation after multiplications, whereas the TMS32010 implementations utilize 32-bit accumulation and two's-complement truncation after accumulations. Also, different coefficient wordlengths were used, 13 bits for μ PD7720 and 16 bits for TMS32010. In the μ PD7720 implementations, the coefficients were stored in the ROM memory, whereas RAM memory was used for the coefficients in the TMS32010. The assembly listings for the first- and second-order WDF all-pass sections of Fig. 11 are given in Appendix C for both processors.

The first-order WDF section was implemented using the three different adaptor types of Figs. 2(a), 2(c), and 3(b). From Table II we can see that the adaptor structures of Figs. 2(c) and 3(b) are as fast (using μ PD7720) or only slightly slower (using TMS32010 or 32020) to implement than the traditional structure of Fig. 2(a). It should also be remembered that overflow control is more difficult to implement in the structure of Fig. 2(a) because a guard-

TABLE II
MINIMUM SAMPLING PERIODS OF DIFFERENT FIRST-ORDER ALL-PASS SECTIONS

Structure	Number of Instruction Cycles	
	μ PD7720	TMS32010 / 32020
Wave Digital		
Adaptor of Fig. 2(a)	6	11 / 11
Adaptor of Fig. 3(b)	6	13 / 11
Adaptor of Fig. 2(c)	6	12 / 12
Mitra-Hirano Type 1A	6	9 / 9
Direct-Form 1	5	8 / 8

TABLE III
MINIMUM SAMPLING PERIODS OF DIFFERENT SECOND-ORDER ALL-PASS SECTIONS

Structure	Number of Instruction Cycles	
	μ PD7720	TMS32010 / 32020
Wave Digital		
Adaptors of Fig. 2(c)	11	24 / 23
Adaptors of Figs. 2(c) and 3(b)	12	25 / 22
Mitra-Hirano Type 3A	9-10*	16-17* / 16-17*
Direct-Form 1	8-9*	13-15* / 12-14*

* Depending on coefficient values.

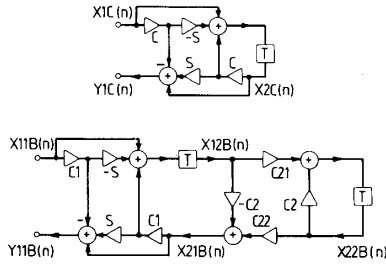


Fig. 11. First- and second-order WDF all-pass sections.

bit is needed in the internal computations. In the second-order case, the adaptor structure of Fig. 2(c) is used as the inner adaptor, and the outer adaptor is of the type of Fig. 2(c) or Fig. 3(b) depending on whether scaling is required or not.

From the Mitra-Hirano structures, only the fastest, 1A and 3A, have been implemented. Even though the Mitra-Hirano structures contain the minimum number of multipliers, they are not faster to implement than the direct-form structures. In fact, the direct-form 1 structures are faster to implement than the related Mitra-Hirano structures 1A_i and 3D. From Table II we can see that using μ PD7720, the direct-form 1 section, takes only one instruction cycle less than the WDF structure, and using TMS32010 (32020) it takes 3-5 (3-4) instructions less. As seen from Table III, the differences are more significant in the second-order case. Especially, in the TMS32010 implementation, the direct-form 1 section is almost twice as fast as the WDF structures.

In practice, saturation arithmetic should also be used. In the case of WDF all-pass structures, this guarantees

stable forced response and smooth behavior during overflow situations. Even though saturation arithmetic does not usually guarantee stable forced response for direct-form and Mitra-Hirano structures, it always improves the behavior during overflow situations. The implementation of saturation arithmetic increases somewhat the execution times of the all-pass sections from those given in Tables II and III for both processors.

VI. EXAMPLE

In this example, we consider the implementation of a complete bandpass filter, including saturation arithmetic, using both μ PD7720 and TMS32010 signal processors. The specifications, which are taken from a practical application, are as follows:

sampling rate: 16 kHz
passband edges: 100 Hz, 5 kHz
passband ripple: 0.2 dB
stopband edges: 62.5 Hz, 6 kHz
stopband attenuation: 46 dB.

The filter is realized as a cascade arrangement of fifth-order elliptic high-pass and low-pass filters. The L_∞ scaling norm was considered to be more appropriate for this application. The resulting uncorrelated noise gains using different types of all-pass sections are compared in Table IV. Because the high-pass part of the frequency response is extremely wide-band (the normalized passband edge is 0.0125π), the WDF structures result in significantly lower noise gains than the direct-form 1 all-pass structures and the best Mitra-Hirano structures. In the high-pass stage, only the WDF all-pass sections give sufficient dynamic range and, due to better stability properties, they are used also in the low-pass stage.

The signal flow diagram of the bandpass filter is shown in Fig. 12. Our choice for the adaptor structures is to use the structure of Fig. 3(b) as the outer adaptor in second-order sections when it is necessary for obtaining good scaling, otherwise we use the structure of Fig. 2(c) with appropriate sign parameters. The latter selection is made because, with $C = 1$, the four-multiplier structure contains a coefficient whose absolute value is larger than unity. On the other hand, when scaling is needed, the resulting coefficient values in the four-multiplier structure are less than one in magnitude. In the low-pass stage, all adaptors are of the type of Fig. 2(c), and proper choice of the sign parameters S gives satisfactory scaling with scaling norms in the range 0.63-1. In the high-pass part, scaling of the end branches of the second-order sections is necessary, and adaptors of the type of Fig. 3(b) are used. The scaling coefficients were optimized as described in Section III-A, and the resulting L_∞ scaling norms are in the range 0.9-1.

This filter was implemented using both μ PD7720 and TMS32010 signal processors. In both cases, the RAM memory was used for storing the coefficients which were quantized to 1 + 15 bit wordlength. In case of μ PD7720, we use 16-bit accumulation of the inner products, corre-

TABLE IV
NOISE GAINS FOR THE BANDPASS FILTER USING DIFFERENT ALL-PASS
STRUCTURES

Structure	Noise Gain in dB	
	Rounding after Accumulations	Rounding after Multiplications
Wave Digital	24.0	27.0
Best Mitra-Hirano	47.5	50.6
Direct-Form I	41.5	47.5

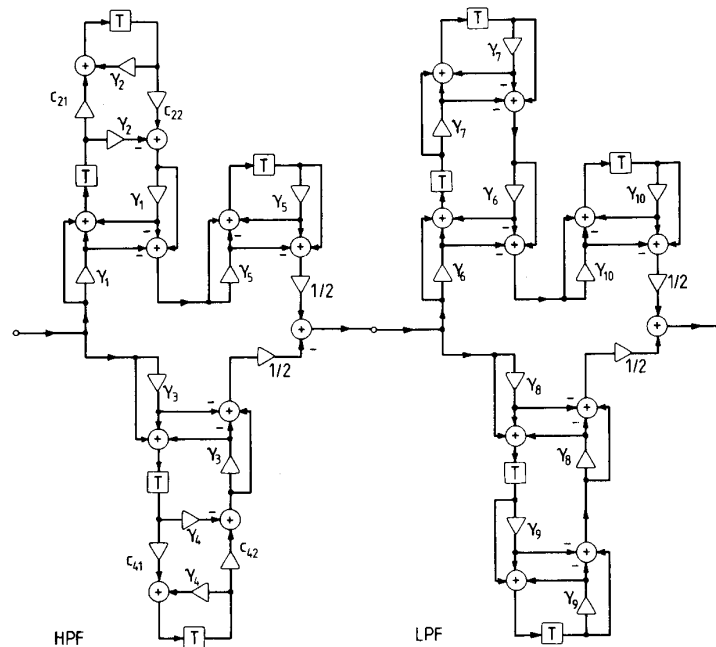


Fig. 12. Signal flow graph for the example design.

sponding to 2's-complement truncation to 1 + 15 bits after multiplications. In TMS32010, we use 32-bit accumulation with 2's-complement truncation to 1 + 15 (or 1 + 14) bits after accumulation. Consequently, the roundoff noise level is about 3 dB lower (or 3 dB higher) in the TMS32010 implementation than in the μ PD7720 implementation.

The observed maximum limit cycle amplitude is about 130 quantization levels peak to peak. With 16-bit data wordlength, this corresponds to about a 54 dB ratio between maximum sinusoidal signal level and maximum limit cycle level.

Saturation arithmetic is used in both implementations in order to guarantee stable forced response and to obtain smooth response during arithmetic overflows. Since the inner products in adaptor computations contain at most 3 terms, the special instructions of μ PD7720 can be used for implementing saturation arithmetic. In this processor, the overflow detection and saturation arithmetic takes two or one instruction cycles depending on whether an overflow has occurred or not. When estimating the worst case sampling period, we can make use of the fact that

overflows cannot occur in every inner product during the same computation period. Especially, there cannot be an overflow in both outputs of a two-port adaptor simultaneously. Furthermore, in the adaptor structures of Fig. 2, with proper choice of the sign parameter, overflows can never appear at output b_2 , and overflow detection is not needed there. This means that, in case of μ PD7720, we have to reserve three cycles for overflow checking and correction when implementing the adaptor of Fig. 3(b), and two cycles when implementing the adaptor of Fig. 2(c). Using μ PD7720, the execution of the filter algorithm takes 120 instruction cycles which corresponds to a maximum sampling rate of 33.333 kHz.

In the case of TMS32010, an extra bit is used during the accumulation so that no overflows may occur during the additions in the adaptor structures. The accumulated result must be left-shifted by one bit before it can be stored. Overflows may occur due to this shift but, unfortunately, the overflow mode does not provide saturation during shift operations. However, this shifting is equivalent to adding the contents of the accumulator to itself. Doing the shift operation in this way, the overflow mode

provides the right type of saturation characteristic in the adaptor implementation [39]. If this addition is performed with full 32-bit accuracy, four extra instruction cycles are required. If only the upper half of the accumulator is added, two extra instruction cycles are required, but in this case we lose one bit in accuracy, i.e., this corresponds to 2's-complement truncation to 1 + 14 bits. As in the case of μPD7720 , the saturation characteristic has to be implemented twice in the adaptor structure of Fig. 3(b) and once in the structure of Fig. 2(c). The number of extra instruction cycles for the overall filter is 48 or 24 depending on whether full-precision additions are used or not.

The overall TMS32010 implementation takes either 160 or 184 instruction cycles, corresponding to 31.25 and 27.77 kHz maximum sampling rates, using 1 + 14 bit or 1 + 15 bit truncation, respectively.

VII. CONCLUSIONS

In this paper, we have compared different first- and second-order all-pass filters with respect to finite wordlength properties and speed of signal processor implementation. When comparing the maximum sampling rate of the different types of all-pass sections, it was observed that the direct-form 1 structures are the fastest among the structures included in the comparison. In the worst case, the WDF structures result in about 50 percent lower sampling rate. However, in certain cases, such as very wide-band or very narrow-band and low-pass/high-pass filters, the WDF structures give superior roundoff noise performance, and it may be impossible to obtain sufficient dynamic range using any other structure. With moderate bandwidths, the direct-form 1 structures and the best Mitra-Hirano structures give often lower roundoff noise power but the differences are less significant. An important advantage of the wave digital all-pass structures is their good stability properties, especially guaranteed stable forced response using saturation arithmetic.

APPENDIX A

ROUND OFF NOISE ANALYSIS OF FIRST-ORDER WDF SECTIONS

Using the adaptor structure of Fig. 2(a) in a first-order all-pass section, the transfer function from the adaptor coefficient to the output is

$$G(z) = \frac{S + z^{-1}}{1 - \gamma z^{-1}} = S + \frac{(1 + S\gamma)z^{-1}}{1 - \gamma z^{-1}}. \quad (\text{A1})$$

The output roundoff noise power is thus

$$\sigma_1^2 = 1 + \frac{(1 + S\gamma)^2}{1 - \gamma^2} \sigma_0^2 = 1 + \frac{1 + S\gamma}{1 - S\gamma} \sigma_0^2. \quad (\text{A2})$$

This is equivalent to (10a) with $C = 1$ and $\nu = 1$.

Using the adaptor structure of Fig. 2(c) with rounding after multiplications, the noise transfer functions are

$$\begin{aligned} G_1(z) &= -\frac{1 + Sz^{-1}}{1 - \gamma z^{-1}} \\ G_2(z) &= \frac{S + z^{-1}}{1 - \gamma z^{-1}}. \end{aligned} \quad (\text{A3})$$

Each of these noise sources contributes to the output noise variance the same amount as given by (A2). Therefore, in this case, (10a) is valid with $C = 1$ and $\nu = 2$.

Let us consider now a first-order all-pass section using the adaptor structure of Fig. 2(b) and quantization after multiplications. The quantization error sequences, e.g., after the multipliers $1 - S\gamma$ and $-\gamma$ are clearly correlated. Assuming that the quantization characteristic satisfies

$$Q[-x] = -Q[x], \quad (\text{A4})$$

we can derive the adaptor output samples as follows:

$$\begin{aligned} b_1(n) &= Q[-\gamma a_1(n)] + Q[(1 + S\gamma) a_2(n)] \\ &= -Q[\gamma a_1(n)] + SQ[\gamma a_2(n)] + a_2(n) \\ b_2(n) &= Q[(1 - S\gamma) a_1(n)] + Q[\gamma a_2(n)] \\ &= a_1(n) - SQ[\gamma a_1(n)] + Q[\gamma a_2(n)]. \end{aligned} \quad (\text{A5})$$

This means that the adaptor output signals are exactly the same as in the structure of Fig. 2(c) with quantization after multiplications. In practice, two's-complement rounding is used, and the property (A4) is not satisfied when the quantization error is exactly half of the quantization step. However, this does not effect the estimate for the output noise power, and we can see that also in this case (10a) is valid with $C = 1$ and $\nu = 2$.

Finally, we consider the first-order all-pass sections based on the adaptor structures of Fig. 2 with rounding of the adaptor output signals. Using the structure of Fig. 2(a), it can be shown that rounding of the adaptor outputs is equivalent to rounding after the adaptor coefficient except for the case where the quantization error is exactly half of the quantization step. With this quantization strategy, all adaptor types of Fig. 2 are equivalent. Therefore, we can conclude that the uncorrelated output roundoff noise power is given by (10a) with $C = 1$ and $\nu = 1$.

APPENDIX B

SCALING AND ROUND OFF NOISE ANALYSIS OF SECOND-ORDER WDF ALL-PASS SECTIONS

In the structure of Fig. 5, the scaling transfer function from input to the delay between the adaptors is

$$F_1(z) = C_1(1 - S_1\gamma_1) \tilde{F}_1(z) \quad (\text{B1})$$

where

$$\begin{aligned} \tilde{F}_1(z) &= \frac{1 - \gamma_2 z^{-1}}{1 + \gamma_2(\gamma_1 - 1)z^{-1} - \gamma_1 z^{-2}} \\ &= \frac{1}{1 - \gamma_1 z^{-1} \left(\frac{-\gamma_2 + z^{-1}}{1 - \gamma_2 z^{-1}} \right)}. \end{aligned} \quad (\text{B2})$$

The maximum of the corresponding amplitude response is thus

$$\|\tilde{F}_1\|_\infty = \frac{1}{1 - |\gamma_1|}. \quad (B3)$$

Including the scaling factor from (B1), (13a) is obtained. From (B2) it follows that

$$\tilde{F}_1(z) = 1 + \gamma_1 z^{-1} \frac{-\gamma_2 + z^{-1}}{1 - \gamma_2 z^{-1}} \tilde{F}_1(z) \quad (B4)$$

and

$$\|\tilde{F}_1\|_2^2 = 1 + \gamma_1^2 \|\tilde{F}_1\|_2^2. \quad (B5)$$

Solving (B5) for $\|\tilde{F}_1\|_2^2$ and including the scaling factor from (B1), (12a) is obtained.

The scaling transfer function from input to the outer delay is

$$F_2(z) = C_1 C_2 (1 - S_1 \gamma_1) (1 - S_2 \gamma_2) z^{-1} \tilde{F}_2(z) \quad (B6)$$

where

$$\begin{aligned} \tilde{F}_2(z) &= \frac{1}{1 + \gamma_2(\gamma_1 - 1)z^{-1} - \gamma_1 z^{-2}} \\ &= \frac{1}{1 + b_1 z^{-1} + b_2 z^{-2}}. \end{aligned} \quad (B7)$$

The L_2 scaling norm corresponding to this all-pole transfer function is

$$\begin{aligned} \|\tilde{F}_2\|_2^2 &= \frac{1 + b_2}{(1 - b_2)((1 + b_2)^2 - b_1^2)} \\ &= \frac{1}{(1 - \gamma_1^2)(1 - \gamma_2^2)} \end{aligned} \quad (B8)$$

and (12b) is obtained by including the gain factor from (B6).

In order to find the maximum value of the amplitude response corresponding to (B6), we locate the minimum value of the magnitude squared response corresponding to the denominator polynomial

$$\begin{aligned} |D(e^{j\omega})|^2 &= D(z) D(z^{-1}) \Big|_{z=e^{j\omega}} \\ &= (1 - b_2)^2 + b_1^2 + 2b_1(1 + b_2) \\ &\quad \cdot \cos(\omega) + 4b_2 \cos^2(\omega). \end{aligned} \quad (B9)$$

Regarding (B9) as a function of $\cos(\omega)$, the abscissa of the extremum point is

$$\cos(\omega_0) = -\frac{b_1(1 + b_2)}{4b_2}. \quad (B10)$$

If now

$$b_2 > 0 \quad (B11a)$$

and

$$\left| \frac{b_1(1 + b_2)}{4b_2} \right| \leq 1, \quad (B11b)$$

then the minimum of the magnitude squared function is achieved at this point and

$$|D(e^{j\omega})|_{\min}^2 = (1 - b_2)^2 + b_1^2 - \frac{b_1^2(1 + b_2)^2}{4b_2}. \quad (B12)$$

Otherwise, the minimum is achieved at $\omega = 0$ or at $\omega = \pi$, and the minimum value is

$$|D(e^{j\omega})|_{\min}^2 = (1 + b_2 - |b_1|)^2. \quad (B13)$$

Equations (13b) and (13c) are now obtained by expressing b_1 and b_2 in terms of γ_1 and γ_2 in (B11)–(B13), and by including the gain factor from (B6).

The noise transfer functions from the multipliers to the output are

$$G_{-\gamma_1} = 1 \quad (B14a)$$

$$G_{\gamma_1} = \frac{(1 + S_1 \gamma_1) z^{-1} (-\gamma_2 + z^{-1})}{C_1 (1 - \gamma_2 z^{-1})} \tilde{F}_1(z) \quad (B14b)$$

$$G_{-\gamma_2} = (1 + S_1 \gamma_1) \tilde{F}_1(z) / C_1 \quad (B14c)$$

$$G_{\gamma_2} = (1 + S_1 \gamma_1) (1 + S_2 \gamma_2) z^{-2} \tilde{F}_2(z) / (C_1 C_2). \quad (B14d)$$

Notice that the noise transfer functions $G_{-\gamma_2}$ and G_{γ_2} have the same forms as (B1) and (B6), only the gain factors and the delay factor of G_{γ_2} are different. In addition, G_{γ_1} and $G_{-\gamma_2}$ differ only by an all-pass factor. Therefore, the expression for output noise power, (14), can be obtained easily by modifying the L_2 norms of the scaling transfer functions.

APPENDIX C ASSEMBLY LISTINGS

First-Order WDF All-Pass Section of Fig. 11(a)

μ PD7720 Implementation

WDF1C:	OP	MOV	@B, A	; /* (ACCB) <-- X1C(N) */
	OP	MOV	@KLR, A	; /* C*X1C(N) STARTS */
	OP	MOV	@A, MEM	; /* (ACCA) <-- X2C(N) */
		SUB	ACCB, M	; /* ADD IF S=-1 */
				; /* (ACCB) <-- (ACCB) - C*X1C(N) */
	OP	MOV	@K, MEM	; /* C*X2C(N) STARTS */
		SUB	ACCA, M	; /* (ACCA) <-- (ACCA) - C*X1C(N) */
	OP	ADD	ACCB, M	; /* (ACCB) <-- (ACCB) + C*X2C(N) */
	OP	MOV	@MEM, B	; /* X2C(N) <-- (ACCB) */
		ADD	ACCA, M	; /* (ACCA) <-- (ACCA) + C*X2C(N) */
				; /* SUB IF S=-1 */
		DPINC		; /* POINT TO NEXT VARIABLE */
		RPDEC		; /* POINT TO NEXT COEFFICIENT */
		RET		; /* RETURN FROM SUBROUTINE */

TMS32010 Implementation

```

WDF1C $MACRO X1C,X2C,Y1C,C,S
      LAC :X2C:,15      (ACC)<-- X2C(N)
      LT :C:            T REG<-- C
      MPY :X1C:          C*X1C(N) STARTS
      SPAC              (ACC)<-- X2C(N)-C*X1C(N)
      MPY :X2C:          C*X2C(N) STARTS
      $IF S,V=1
      APAC              (ACC)<-- X2C(N)-C*X1C(N)+C*X2C(N)
      $ELSE
      SPAC
      $ENDIF
      SACH :Y1C:,1      Y1C<-- (ACC)
      PAC              (ACC)<-- C*X2C(N)
      MPY :X1C:          C*X1C(N) STARTS
      $IF S,V=1
      SPAC              (ACC)<-- C*X2C(N)-C*X1C(N)
      $ELSE
      APAC
      $ENDIF
      ADD :X1C:,15      (ACC)<-- X1C(N)+C*X2C(N)-C*X1C(N)
      SACH :X2C:,1      X2C<-- (ACC)
      $END

```

*Second-Order WDF All-Pass Section of Fig. 11(b)**μPD7720 Implementation*

```

WDF2B: OP XOR ACCB,B ;/*CLEAR ACCB */
      OP MOV @TR,A /*TR<--X11B(N) */
      XOR ACCA,IDB ;/*CLEAR ACCA */
      OP MOV @KLR,MEM ;/*C21*X12B(N) STARTS */
      RPDEC ;/*POINT TO NEXT COEFFICIENT*/
      OP MOV @L,RO /* C2*X12B(N) STARTS */
      ADD ACCB,M /*(ACCB)<--C21*X12B(N)*/
      M1 ;/*POINT TO X22B(N)*/
      OP MOV @K,MEM /*C2*X22B(N) STARTS*/
      SUB ACCA,M /*(ACCA)<-- -C2*X12B(N)*/
      RPDEC ;/*POINT TO NEXT COEFFICIENT*/
      OP MOV @L,RO /* C22*X22B(N) STARTS */
      ADD ACCB,M /*(ACCB)<--(ACCB)+C2*X22B(N)*/
      RPDEC ;/*POINT TO NEXT COEFFICIENT*/
      OP MOV @MEM,B /* X22B(N)<--(ACCB) */
      ADD ACCA,M /*(ACCA)<--(ACCA)+C22*X22B(N)*/
      M1 ;/*POINT TO X12B(N)*/
      OP MOV @KLR,A ;/* C1*X21B(N) STARTS */
      OP MOV @B,TR /*(ACCB)<--X11B(N) */
      SUB ACCA,M ;/*(ACCA)<--(ACCA)-C1*X21B(N)*/
      OP MOV @K,TR /* C1*X11B(N) STARTS */
      ADD ACCB,M ;/*(ACCB)<--(ACCB)+C1*X21B(N)*/
      OP ADD ACCB,M ;/*(ACCB)<--(ACCB)+C1*X11B(N)*/
      OP MOV @MEM,B /* X12B(N)<--(ACCB) */
      SUB ACCA,M /*(ACCA)<--(ACCA)-C1*X11B(N)*/
      DPINC ;/*POINT TO NEXT VARIABLE*/
      RPDEC ;/*POINT TO NEXT COEFFICIENT*/
      RET ;/*RETURN FROM SUBROUTINE*/

```

TMS32010 Implementation

```

WDF2B $MACRO X11B,X12B,X21B,X22B,Y11B,C21,C22,C1,C2
      LT :C22:          T REG<-- C22
      MPY :X22B:         C22*X22B(N) STARTS
      PAC              (ACC)<-- C22*X22B(N)
      LT :C2:           T REG<-- C2
      MPY :X12B:         C2*X12B(N) STARTS
      SPAC              (ACC)<-- C22*X22B(N)-C2*X12B(N)
      SACH :X21B:,1      X21B<-- (ACC)
      MPY :X22B:         C2*X22B(N) STARTS
      PAC              (ACC)<-- C2*X22B(N)
      LT :C21:          T REG<-- C21
      MPY :X12B:         C21*X12B(N) STARTS
      APAC              (ACC)<-- C2*X22B(N)+C21*X12B(N)
      SACH :X22B:,1      X22B(N)<-- (ACC)
      LAC :X21B:,15      (ACC)<-- X21B(N)
      LT :C1:           T REG<-- C1
      MPY :X21B:         C1*X21B(N) STARTS

```

```

SPAC                                (ACC)<-- X21B(N)-C1*X21B(N)
MPY : X11B:                        C1*X11B(N) STARTS
SPAC                                (ACC)<-- X21B(N)-C1*X21B(N)-C1*X11B(N)
SACH : Y11B:, 1                    Y11B(N)<-- (ACC)
PAC                                (ACC)<-- C1*X11B(N)
ADD : X11B:, 15                    (ACC)<-- X11B(N)+C1*X11B(N)
MPY : X21B:                        C1*X21B(N) STARTS
AFAC                                (ACC)<-- X11B(N)+C1*X11B(N)+C1*X21B(N)
SACH : X12B:, 1                    X12B<-- (ACC)
#END

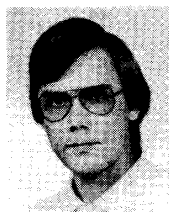
```

ACKNOWLEDGMENT

The authors are grateful to E. Kyriakis-Bitaros, a graduate student at the Department of Physics, Patras University, for his help in writing the programs for TMS32010.

REFERENCES

- [1] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1975.
- [2] R. Nouta, "The Jaumann structure in wave-digital filters," *Int. J. Circuit Theory and Appl.*, vol. 2, pp. 163-174, June 1974.
- [3] A. Fettweis, H. Levin, and A. Sedlmeyer, "Wave digital lattice filters," *Int. J. Circuit Theory and Appl.*, vol. 2, pp. 203-211, June 1974.
- [4] L. Gazsi, "Explicit formulas for lattice wave digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-32, pp. 68-88, Jan. 1985.
- [5] R. Ansari and B. Liu, "A class of low-noise computationally efficient recursive digital filters with applications to sampling rate alterations," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 90-97, Feb. 1985.
- [6] P. P. Vaidyanathan, S. K. Mitra, and Y. Neuvo, "A new approach to the realization of low-sensitivity IIR digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 350-361, Apr. 1986.
- [7] T. Saramäki, "On the design of digital filters as a sum of two all-pass filters," *IEEE Trans. Circuits Syst.*, vol. CAS-32, pp. 1191-1193, Nov. 1985.
- [8] W. Wegener, "Wave digital directional filters with reduced number of multipliers and adders," *Arch. Elek. Übertragung.*, vol. 33, pp. 239-243, June 1979.
- [9] J. A. Nossek and H.-D. Schwartz, "Wave digital lattice filters with applications in communication systems," in *Proc. IEEE Int. Symp. Circuits Syst.*, Newport Beach, CA, May 1983, pp. 845-848.
- [10] R. A. Valenzuela and A. G. Constantinides, "Digital signal processing schemes for efficient interpolation and decimation," *IEE Proc. G, Electron. Circuits and Syst.*, vol. 130, pp. 225-235, Dec. 1983.
- [11] R. Ansari and B. Liu, "Efficient sampling rate alteration using recursive (IIR) digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 1366-1373, Dec. 1983.
- [12] A. Fettweis, J. A. Nossek, and K. Meerkötter, "Reconstruction of signals after filtering and sampling rate reduction," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 893-902, Aug. 1985.
- [13] L. Taxen, "Polyphase filter banks using wave digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, pp. 423-428, June 1981.
- [14] M. Renfors and T. Saramäki, "Recursive Nth-band digital filters—Part I: Design and properties, Part II: Design of multistage decimators and interpolators," *IEEE Trans. Circuits Syst.*, vol. CAS-34, pp. 24-51, Jan. 1987.
- [15] W. Drews and L. Gazsi, "A new design method for polyphase filters using all-pass sections," *IEEE Trans. Circuits Syst.*, vol. CAS-33, pp. 346-348, Mar. 1986.
- [16] L. Gazsi, "Single chip filter bank with wave digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, pp. 709-718, Oct. 1982.
- [17] S. K. Mitra, Y. Neuvo, and P. P. Vaidyanathan, "Complementary IIR digital filter banks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Tampa, FL, Mar. 1985, pp. 529-532.
- [18] R. Ansari and B. Liu, "Transmultiplexer design using all-pass filters," *IEEE Trans. Commun.*, vol. COM-30, pp. 1569-1574, July 1982.
- [19] W. Drews, "Transmultiplexer with all-pass filters—A new design method and its passive realization," in *Proc. Euro. Conf. Circuit Theory and Design*, Prague, Czechoslovakia, Sept. 1985, pp. 383-386.
- [20] M. Renfors, Y. Neuvo, and S. K. Mitra, "A new class of uniform filter banks based on recursive Nth-band filters," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Tokyo, Japan, Apr. 1986, pp. 2555-2558.
- [21] M. Renfors and T. Saramäki, "A class of approximately linear phase digital filters composed of all-pass subfilters," in *Proc. IEEE Int. Symp. Circuits Syst.*, San Jose, CA, May 1986, pp. 678-681.
- [22] C. W. Kim and R. Ansari, "Approximately linear phase IIR filters using all-pass sections," in *Proc. IEEE Int. Symp. Circuits Syst.*, San Jose, CA, May 1986, pp. 661-664.
- [23] P. P. Vaidyanathan, P. A. Regalia, and S. K. Mitra, "Design of doubly-complementary IIR digital filters using a single complex all-pass filter, with multirate applications," *IEEE Trans. Circuits Syst.*, vol. CAS-34, pp. 378-389, Apr. 1987.
- [24] T. A. C. M. Claassen, W. F. G. Mecklenbräuker, and J. B. H. Peek, "On the stability of the forced response of digital filters with overflow nonlinearities," *IEEE Trans. Circuits Syst.*, vol. CAS-22, pp. 692-696, Aug. 1975.
- [25] A. Fettweis, "Wave digital filters: Theory and practice," *Proc. IEEE*, vol. 74, pp. 270-327, Feb. 1986.
- [26] *μPD 7720 SPI and EVAKIT-7720 Operation Manuals*, NEC Microcomputers.
- [27] *TMS32010 User's Guide*, Texas Instruments Inc., 1986.
- [28] G. Lucioni, "General processor application; CAD tool for filter design," in *Proc. Euro. Signal Processing Conf.*, The Hague, The Netherlands, Sept. 1986, pp. 1271-1274.
- [29] S. K. Mitra and K. Hirano, "Digital all-pass networks," *IEEE Trans. Circuits Syst.*, vol. CAS-21, pp. 688-700, Sept. 1974.
- [30] T. A. C. M. Claassen, W. F. G. Mecklenbräuker, and J. B. H. Peek, "Effects of quantization and overflow in recursive digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 517-529, Dec. 1976.
- [31] A. Fettweis and K. Meerkötter, "On adaptors for wave digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 516-525, Dec. 1975.
- [32] —, "Suppression of parasitic oscillations in wave digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-22, pp. 239-246, Mar. 1975.
- [33] L. B. Jackson, "On the interaction of roundoff noise and dynamic range in digital filters," *Bell Syst. Tech. J.*, vol. 49, pp. 159-184, Feb. 1970.
- [34] M. Renfors and E. Zigouris, "Signal processor implementation of lattice wave digital filters," in *Proc. Int. Conf. Digital Signal Processing*, Florence, Italy, Sept. 1984, pp. 233-237.
- [35] E. I. Jury, *Theory and Application of the z-Transform Method*. New York: Wiley, 1964.
- [36] A. H. Gray, Jr., and J. D. Markel, "A normalized digital filter structure," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 268-277, June 1975.
- [37] P. M. Ebert, J. E. Mazo, and M. G. Taylor, "Overflow oscillations in digital filters," *Bell Syst. Tech. J.*, vol. 48, pp. 2999-3020, Nov. 1969.
- [38] K. T. Erickson and A. N. Michel, "Stability analysis of fixed-point digital filters using computer generated Lyapunov functions—Part I: Direct form and coupled form filters," *IEEE Trans. Circuits Syst.*, vol. CAS-32, pp. 113-132, Feb. 1985.
- [39] K. Valkonen, "Design of narrowband digital filters for sampling rate reduction," M.Sc. thesis (in Finnish), Tampere Univ. Technol., Finland, Apr. 1987.



Markku Renfors (S'77-M'82) was born in Suoniemi, Finland, on January 21, 1953. He received the Diploma Engineer, Licentiate of Technology, and Doctor of Technology degrees in electrical engineering from the Tampere University of Technology, Finland, in 1978, 1981, and 1982, respectively.

Since 1977 he has been with the Department of Electrical Engineering at the Tampere University of Technology holding various research positions. He was a Visiting Research Scholar at the Lin-

köping University, Sweden, in 1983, and at the University of California, Santa Barbara, in Spring 1985 and in Autumn 1986. His research interests are in digital signal processing, especially in structures and implementation of digital filters.



Evangelos Zigouris was born in Agrinion, Greece, on June 13, 1948. He received the B.Sc. degree (with honors) and the Ph.D. degree, both in physics, from the University of Patras, Patras, Greece.

Since January 1975 he has been with the Electronics Laboratory, Department of Physics, University of Patras, as a Research and Teaching Assistant, where he is presently a Lecturer. From October 1982 until February 1984 he was a Visiting Engineer at the Department of Electrical En-

gineering, Division of Applied Electronics, University of Linköping, Linköping, Sweden. His current research interests are in one- and two-dimensional digital signal processing and digital signal processor and microprocessor applications.