Realization of Digital Filters Using Block-Floating-Point Arithmetic

ALAN V. OPPENHEIM, Member, IEEE
Massachusetts Institute of Technology
Lincoln Laboratory
Lexington, Mass. 02173
and
Department of Electrical Engineering, and
Research Laboratory of Electronics
Cambridge, Mass.

Abstract
Recently, statistical models for the effects of roundoff noise in fixed-point and floating-point realizations of digital filters have been proposed and verified, and a comparison between these realizations presented. In this paper a structure for implementing digital filters using block-floating-point arithmetic is proposed and a statistical analysis of the effects of roundoff noise is carried out. On the basis of this analysis, block-floating-point is compared to fixed-point and floating-point arithmetic with regard to roundoff noise effects.

A Structure for Block-Floating-Point Realization
In block-floating-point arithmetic the input and filter states (i.e., the outputs of the delay registers) are jointly normalized before the multiplications and adds are performed using fixed-point arithmetic. The scale factor obtained during the normalization is then applied to the final output to produce a fixed-point result. To illustrate, consider a first-order filter described by the difference equation

\[ y_n = x_n + a_1 y_{n-1}. \]  

(1)

For convenience we will treat all numbers as fixed-point fractions. To perform the computation in a block-floating-point manner, we define

\[ A_n = \frac{1}{\text{IP}[\max \{ |x_n|, |y_{n-1}| \}]} \]  

(2)

where IP[M] is used to denote the integer power of two such that \( M < \text{IP}(M) \leq 2M \), i.e., with \( M \) written as \( 2^m \cdot P \) with \( P \) between \( \frac{1}{2} \) and 1, \( \text{IP}(M) = 2^m \). For \( M \) a fraction, \( 2^m \) is less than or equal to unity so that \( A_n \) is greater than or equal to unity. Thus \( A_n \) represents the power-of-two scaling which will jointly normalize \( x_n \) and \( y_{n-1} \). Thus with block-floating-point we can compute \( y_n \) as

\[ y_n = \frac{1}{A_n} \left( A_n x_n + a_1 A_n y_{n-1} \right) \]  

(3)

where the multiplications and addition in (3) are carried out in a fixed-point manner.
Because of the recursive nature of the computation for a digital filter, it is advantageous to modify (3) as

\[ y_n = A_n x_n + a_1 \Delta_n w_{1n} \]  

(4)

with

\[ w_{1n} = A_{n-1} y_{n-1} \]

\[ y_n = \frac{1}{A_n} \hat{y}_n \]

and

\[ \Delta_n = A_n / A_{n-1}. \]

The difference between (3) and (4) is meant to imply that the number \( A_n y_n \) rather than \( y_n \) is stored in the delay register of the filter. Because of (2), \( A_n y_n \) is always more accurate (or as accurate) as \( y_n \) since multiplication by \( A_n \) corresponds to a left shift of the register.

A disadvantage with (4) is that \( y_{n-1} \) must be available to compute \( A_n \), and \( A_n \) must then be obtained from \( A_n \) and \( A_{n-1} \). An alternative is represented by the set of equations

\[ \hat{y}_n = \Delta_n \hat{x}_n + a_1 \Delta_n w_{1n} \]  

(5a)

with

\[ \hat{x}_n = A_{n-1} x_n \]  

(5b)

and

\[ \Delta_n = A_n / A_{n-1} = \frac{1}{\text{IP}[\max \{|\hat{x}_n|, |w_{1n}|\}]}. \]  

(5c)

In this case, we first scale \( x_n \) by \( A_{n-1} \) to form \( \hat{x}_n \) and then determine the incremental scaling using (5c). As in (4), the scaled value \( \hat{y}_n \) is stored in the delay register and the output value \( y_n \) is determined from \( \hat{y}_n \). If we consider the general case of an \( N \)th order filter of the form

\[ y_n = x_n + a_1 y_{n-1} + a_2 y_{n-2} + \cdots + a_N y_{n-N}, \]

then the block-floating-point realization corresponding to (5) and represented in the direct form is depicted in Fig. 1.

For the general case,

\[ \Delta_n = \frac{1}{\text{IP}[\max \{|\hat{x}_n|, |w_{1n}|, \cdots, |w_{Nn}|\}]} \]  

(6)

and

\[ A_n = \frac{1}{\text{IP}[\max \{|x_n|, |y_{n-1}|, \cdots, |y_{n-N}|\}]} \]  

(7)

where \( \alpha \) is a constant that may be changed depending on the filter to be implemented. In a first-order filter, for example, \( \alpha \) need never be greater than two.

The Effect of Roundoff Noise in Block-Floating-Point Filters

In evaluating the performance of the block-floating-point realization in the presence of roundoff noise, we will restrict attention to the implementation of (5) and Fig. 1 for the first- and second-order cases. We will assume that no roundoff occurs in the computation of \( \hat{x}_n \) from \( x_n \) and the subsequent multiplication by \( \Delta_n \). Since \( A_{n-1} \) and \( A_n \Delta_n \) are always nonnegative powers of two, that is, they always correspond to a positive scaling, the above assumption corresponds to allowing more bits in the representation of the intermediate variable \( \hat{x}_n \). This is reasonable if we take the attitude that it is primarily in the variables used in the arithmetic computations that the register length is important.

For the first-order case, roundoff noise is introduced in the multiplication of \( w_{1n} \) by \( \Delta_n \), the multiplication by \( a_1 \), and the final multiplication by \( 1/A_n \). The effects of multiplier roundoff will be modeled by representing the round-
off by additive white noise sources. We consider, for convenience, the fixed-point numbers in the registers to represent signed fractions, with the register length excluding sign denoted by \( t \) bits. Each of the roundoff noise generators is assumed to be white, mutually independent and independent of the input, and to have a variance \( \sigma_r^2 \) equal to \((1/12) \cdot 2^{-2t}\). The network for the first-order filter including the noise sources representing roundoff error is presented in Fig. 2(A). In Fig. 2(B) an equivalent representation is shown, where the noise sources are at the filter input. If we consider the input to be a stationary random signal, then the noise source \( \xi_n \) will be white stationary random noise with variance

\[
\sigma_{\xi}^2 = \sigma_r^2 (1 + \alpha_2^2) k_2^2
\]

(8)

where \( k_2^2 \) denotes the expected value of \((1/A_j)^2\). Letting \( \eta_n \) denote the noise in the filter output due to the noise \( \xi_n \), the variance of the output noise \( \eta_n \) will be

\[
\sigma_{\eta}^2 = \sigma_{\xi}^2 \sum_{n=0}^{\infty} h_n^2 + \frac{1}{\alpha_n^2} = \sigma_r^2 \left[ 1 + \frac{1 + \alpha_1^2}{1 - \alpha_1^2} k_2^2 \right].
\]

(9)

This result is derived by observing that in Fig. 2(B) the transmission from the noise source \( \xi_n \) to the output is that of a first-order filter with unit sample response \( h_n \) given by \( h_n = \alpha^n \). For the case of a second-order filter a similar procedure can be followed. Fig. 3(A) shows a second-order filter with the roundoff noise sources included. In Fig. 3(B) an equivalent representation is shown, where equivalent noise sources are introduced at the filter input. Again, considering the input to be a stationary random signal, then

\[
\sigma_{\xi}^2 = \frac{1}{(A_n)^2} \sigma_{\xi}^2 \left[ 4r^2 \cos^2 \theta + 2 + r^4 \right]
\]

\[
+ r^4 \sigma_{\xi}^2 \left[ \frac{1}{(A_{n-1})^2} \right]
\]

(10)

\[
= k_2^2 \sigma_r^2 \left[ 4r^2 \cos^2 \theta + 2 + 2r^4 \right]
\]

where we assume that the mean-square values of \((1/A_n)\) and \((1/A_{n-1})\) are equal. Hence the variance of the output noise \( \eta_n \) is

\[
\sigma_{\eta}^2 = \sigma_{\xi}^2 + k_2^2 \sigma_r^2 G \left[ 4r^2 \cos^2 \theta + 2 + 2r^4 \right]
\]

(11)

where

\[
G = \left( \frac{1 + r^2}{1 - r^2} \right) \frac{1}{1 + r^4 - 4r^2 \cos^2 \theta + 2r^4}.
\]

(12)

**Experimental Verification**

To verify the validity of (9) and (11) the values of \( k_2^2 \) were measured and the values of \( \sigma_{\xi}^2 \) computed from (9) and (11) using these measured values. These results were taken as the theoretical results since they incorporate the
A Comparison of Block-Floating-Point, Floating-Point, and Fixed-Point Realizations

Using the model presented in the previous section, the block-floating-point realization of digital filters can be compared with fixed-point and floating-point realizations. The comparison to be presented here will be on the basis of the output noise-to-signal ratio when the input is a random signal with a flat spectrum, using results presented by Gold and Rader [1], Kaneko and Liu [2], and Weinstein and Oppenheim [3]. With \( \sigma_n^2 \) denoting the variance of the roundoff noise as it appears in the output, we have for the first-order filter

\[
\text{fixed point}: \quad \sigma_n^2 = \frac{1}{12} \cdot 2^{-2t} \frac{1}{1 - a_1^2} \tag{13}
\]

\[
\text{floating point}: \quad \sigma_n^2 = 0.23 \cdot 2^{-2t} \frac{1 + a_1^2}{1 - a_1^2} \sigma_y^2 \tag{14}
\]

and for the second-order filter

\[
\text{fixed point}: \quad \sigma_n^2 = \frac{1}{6} \cdot 2^{-2t} G \tag{15}
\]

\[
\text{floating point}: \quad \sigma_n^2 = 0.23 \cdot 2^{-2t} \left[ 1 + G \left( 3r^4 + 12r^2 \cos^2 \theta - \frac{10r^4 \cos^2 \theta}{1 + r^2} \right) \right] \sigma_y^2 \tag{16}
\]

where \( t \) is the number of bits in the mantissa, not including sign, \( \sigma_y^2 \) is the variance of the output signal, and \( G \) is given by (12). In the fixed-point case the output noise is independent of the output signal variance, and in the floating-point case the output noise is proportional to the output signal variance. The expression for block-floating-point noise has a term independent of the signal and a term which depends on the signal through the factor \( k^2 \). In both the fixed-point and block-floating-point cases, the dynamic range for the output is constrained by the register length. Consequently, as the filter gain increases, the input must be scaled down to prevent the output from overflowing the register length. Since the output is given by

\[ y_n = \sum_{k=0}^{\infty} h_k x_{n-k}, \]

then

\[ |y_n| \leq \max (|x_n|) \sum_{k=0}^{\infty} |h_k|. \]

To insure that the output fits within a register length, we require that, with \( x_n \) and \( y_n \) interpreted as fractions,

\[ |y_n| \leq 1 \]

so that

OPPENHEIM: DIGITAL FILTERS USING BLOCK-FLOATING-POINT ARITHMETIC 133
\[
\frac{-1}{\sum_{k=0}^{\infty} |h_k|} \leq x_n \leq \frac{1}{\sum_{k=0}^{\infty} |h_k|}.
\]

With this constraint on the input, we can then compute an output noise-to-signal ratio for fixed-point, floating-point, and block-floating realizations. Specifically, for the first-order case,

\[
\left(\frac{\sigma_n^2}{\sigma_y^2}\right)_{\text{fixed point}} = \frac{1}{12} 2^{-2t} \frac{3}{(1 - a_1)^2}
\]

\[
\left(\frac{\sigma_n^2}{\sigma_y^2}\right)_{\text{floating point}} = 0.23 \cdot 2^{-2t} \frac{1 + a_1^2}{1 - a_1^2}
\]

\[
\left(\frac{\sigma_n^2}{\sigma_y^2}\right)_{\text{block floating}} = \frac{1}{12} 2^{-2t} \left[ 3(1 - a_1^2) (1 - a_1)^2 + 3(1 + a_1^2) k^2 \right]
\]

where \(k^2\) is the value for \(k^2\) when \(x_n\) is uniformly distributed between plus and minus unity.\(^1\)

In a similar manner, for the second-order case,

\[
\left(\frac{\sigma_n^2}{\sigma_y^2}\right)_{\text{fixed point}} = \frac{1}{2} 2^{-2t} \left( \frac{1}{\sin \theta} \sum_{n=0}^{\infty} r^n \sin [(n + 1) \theta] \right)^2
\]

\[
\left(\frac{\sigma_n^2}{\sigma_y^2}\right)_{\text{floating point}} = (0.23) 2^{-2t} \left[ 1 + G \left( 3r^4 + 12r^2 \cos^2 \theta - 16 \frac{r^4 \cos^2 \theta}{1 + r^2} \right) \right]
\]

\[
\left(\frac{\sigma_n^2}{\sigma_y^2}\right)_{\text{block floating}} = \frac{1}{12} 2^{-2t} \left( \frac{3}{G \sin \theta} \sum_{n=0}^{\infty} r^n \sin [(n + 1) \theta] \right)^2 + 3k^2 (2 + 2r^4 + 4r^2 \cos^2 \theta)
\]

In Fig. 4, (18), (19), and (20) are compared. In Fig. 5, (21), (22), and (23) are compared. In these figures the noise-to-signal ratios are plotted in bits so that the difference between two of the curves reflects the number of bits that the mantissas should differ by to achieve the same noise-to-signal ratio. In each of the cases, the difference between floating-point and block-floating-point is approximately constant as the filter gain (or the proximity of the poles to the unit circle) increases. This difference is approximately one bit in the first-order case and two bits in the second-order case. In contrast, the fixed-point noise-to-signal ratio increases at a faster rate than floating-point or block-floating-point, and for low gain is better and for high gain is worse than block-floating-point.

In evaluating the comparison between fixed-point, floating-point, and block-floating-point filter realizations, it is important to note that Figs. 4 and 5 are based only on the mantissa length and do not reflect the additional bits needed to represent the characteristic in either floating-point or block-floating-point arithmetic.

An additional consideration which is not reflected in these curves is that in both fixed-point and block-floating-point the noise-to-signal ratio is computed on the assump-
tion that the input signal is as large as possible consistent with the requirement that the output fit within the register length. If the input signal is in fact smaller than permitted, then the noise-to-signal ratio for the fixed-point case will be proportionately higher. For block-floating-point, as the input signal decreases, the output noise variance asymptotically approaches $\sigma^2$.

For the case of high-gain filters, (18) through (23) can be approximated by asymptotic expressions which place in evidence the relationship between them. For the high-gain case, that is, for $a_1$ close to unity in the first-order filter and $r$ close to unity and $\theta$ small in the second-order filter, we will assume that $|x_0|$ is always smaller than $|y_0|$ so that $(1/A_n)\approx 2|y_0|$ for the first-order filter and $(1/A_n)\approx 4|y_0|$ for the second-order filter. Then, if we consider $y_n$ as a random variable with a symmetric probability density,

$$k^2 \approx \frac{4}{3} \cdot \frac{1}{1 + a_1^2}$$

in (20) and

$$k^2 \approx \frac{16}{3} \cdot G$$

in (23).

Representing $a$ as $1-\delta$ for the first-order case and $r$ as $1-\delta$ for the second-order case, with $\delta$ small, we can approximate (18) through (20) as

$$2^{2\delta} \left( \frac{\sigma_n^2}{\sigma_y^2} \right)_{\text{fixed point}} \approx \frac{0.25}{\delta^2}$$

in (24)

$$2^{2\delta} \left( \frac{\sigma_n^2}{\sigma_y^2} \right)_{\text{floating point}} \approx 0.23 \cdot \frac{1}{\delta}$$

in (25)

$$2^{2\delta} \left( \frac{\sigma_n^2}{\sigma_y^2} \right)_{\text{block floating}} \approx 0.83 \cdot \frac{1}{\delta}$$

in (26)

For the second-order case we will want to bracket the expression

Fig. 5. Comparison of noise-to-signal ratios for second-order filter using fixed-point, floating-point, and block-floating-point arithmetic.
This sum is the sum of the absolute values of the impulse response and as such is an upper bound on the filter output with a maximum input of unity. Consequently, it must be greater than or equal to the response of the second-order filter to a sinusoid of unity amplitude at the resonant frequency. This resonance response is given by 
\[ \frac{1}{(1-r)(1+r^2-2r \cos 2\theta)^{1/2}} \] or, with the high-gain approximation, 
\[ \frac{1}{(2\delta \sin \theta)^{1/2}} \]. An upper bound is easily obtained on the sum as
\[ \left( \frac{1}{\sin \theta} \sum_{n=0}^{\infty} r^n \right) \left( \frac{1}{\frac{1}{4} + \frac{4(1 + \cos^2 \theta)}{3 \sin^2 \theta}} \right) \leq 2^{2t} \left( \frac{\sigma_n}{\sigma_y^2} \right)^{\text{fixed point}} \leq \frac{7 + \cos^2 \theta}{3 \delta \sin^2 \theta} \] (29)

We note that the behavior of these expressions as a function of \( \delta \) is consistent with the results plotted in Figs. 4 and 5.

**Acknowledgment**

The author would like to thank Dr. C. Weinstein for his contributions to this work through many discussions and Mrs. A. Fried for her help in carrying out many of the computations involved.

**References**


**Alan V. Oppenheim** (S'57–M'65) was born in New York, N. Y., on November 11, 1937. He received the S.B. and S.M. degrees in 1961 and the Sc.D. degree in 1964 in electrical engineering from the Massachusetts Institute of Technology, Cambridge.

From 1961 to 1964 he was a member of the M.I.T. Research Laboratory of Electronics and an Instructor in the Department of Electrical Engineering, where he received a departmental teaching award in 1963. During this period his primary activities centered on system and communication theory. His doctoral research involved the application of modern algebra to the characterization of nonlinear systems. Since 1964 he has been an Assistant Professor in the M.I.T. Department of Electrical Engineering and a Staff Member of the Research Laboratory of Electronics. His present research activities are in the areas of speech communication and digital waveform processing. He is presently on a leave of absence at the M.I.T. Lincoln Laboratory, Lexington, Mass.

Dr. Oppenheim is a member of Tau Beta Pi,Eta Kappa Nu, Sigma Xi, and the Acoustical Society of America.