Chapter 16

Roundoff Noise in IIR Digital Filters

It will not be possible in this brief chapter to discuss all forms of IIR (infinite impulse response) digital filters and how quantization takes place in them. With a few simple examples however, it should be possible for the reader to be able to model the quantization process and to evaluate the effects of quantization noise at the outputs of specific IIR filters that may be of interest.

We assume in this chapter that the coefficients of the filter are infinitely precisely given. In digital implementation, this will not be true – the consequences of imperfect accuracy in the coefficients are discussed in Chapter 21.

16.1 A ONE-POLE DIGITAL FILTER

A simple one-pole digital filter can be represented by the following difference equation.

$$y(k) - ay(k-1) = x(k)\,. \tag{16.1}$$

A block diagram representing this filter is shown in Fig. 16.1.

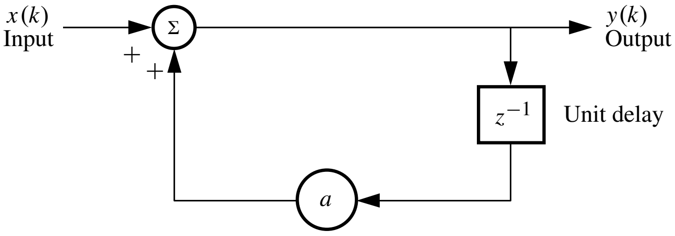

Figure 16.1 Block diagram of one-pole digital filter.

The filter input is $x(k)$, and the filter output is $y(k)$. The discrete time index is k. A delay of one sample time is represented by z^{-1}.

The transfer function of the filter can be obtained by taking the z-transform of the difference equation Eq. (16.1), or by using the feedback formula for the block diagram of Fig. 16.1. The transfer function from input to output is

$$H(z) \overset{\Delta}{=} \frac{Y(z)}{X(z)} = \frac{1}{1 - az^{-1}}\,, \tag{16.2}$$

where $Y(z)$ is the z-transform of $y(k)$ and $X(z)$ is the z-transform of $x(k)$. The impulse response is geometric, with a geometric ratio of a. The filter will be stable when a is chosen so that $|a| < 1$.

16.2 QUANTIZATION IN A ONE-POLE DIGITAL FILTER

Quantization generally takes place within the feedback loop just after multiplication by the coefficient or gain a. Figure 16.2(a) illustrates one way that this might occur. The feedback signal is quantized, and the quantized signal is combined with (added to) the input signal to form the output signal.

If there were no quantization, the number of digits representing the feedback signal would grow with time. Multiplication by the feedback coefficient a adds the number of digits of a to the feedback signal. The feedback signal going round and round the loop would have a number of digits that would grow without bound. In short time, this would exceed the word length of any computer. Quantization is employed to prevent this problem.

There is something unusual in Fig. 16.2(a) however. The quantizer input contains components of its own output. The quantization noise gets back into the quantizer input. The quantization noise will therefore be correlated with the quantizer input. This is contrary to the conditions that we have previously found when the quantization noise has the properties of PQN. Does this mean that PQN conditions can never be met for quantization within a feedback loop? This could be a major concern.

Fig. 16.2(b) represents a feedback system that is identical to that of Fig. 16.2(a). In Fig. 16.2(b), the quantizer is drawn in isolation, so that one could eye its behavior without being confused by its being within a feedback loop. So we put our head inside the isolation chamber and not worry about the rest of the world outside.

Suppose that the first-order PDF of the quantizer input is smooth and spread over many quantum boxes so that the first-order QT II is satisfied to a good approximation. Then the first-order PDF of the quantization noise will be uniform to a good approximation. Suppose next that the high-order QT II is satisfied to a good approximation. Then the high-order PDF of the quantization noise will be uniform to a good approximation. This means that the quantization noise samples will be uncorrelated over time as well as being uniformly distributed. In fact, all multidimensional joint

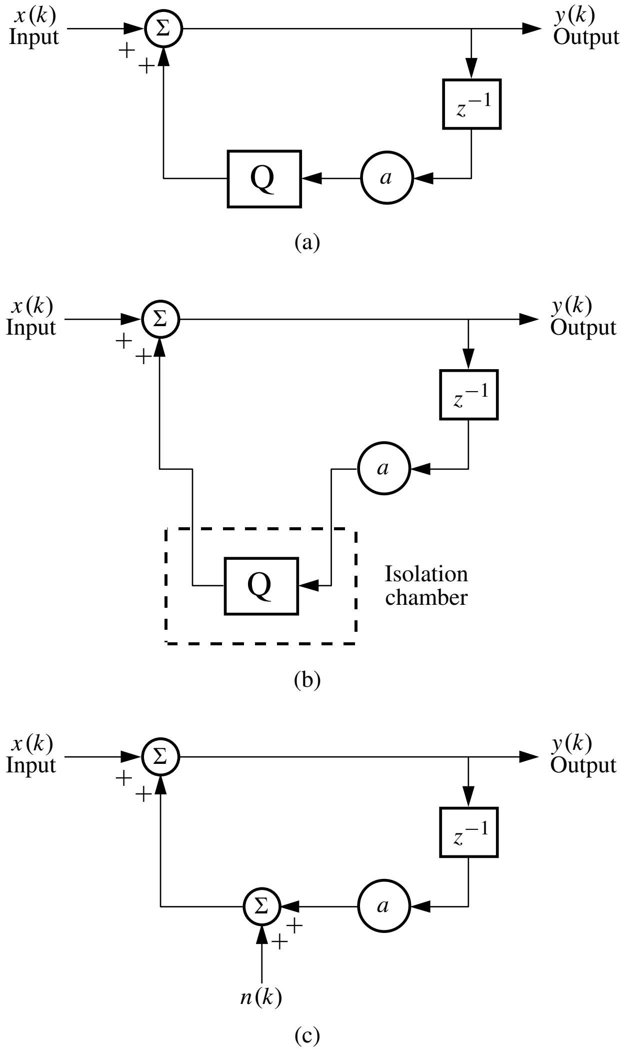

Figure 16.2 Quantization within a feedback loop: (a) an IIR filter with a quantizer within its feedback loop; (b) isolation of the quantizer; (c) PQN model for quantization within the loop.

input/noise/output moments would be the same as if the quantizer were replaced by a source of additive PQN, uniformly distributed noise that is independent from sample to sample over time and generated independently of all other signals. In practice, these conditions are normally met to a very close approximation.

Assuming that the quantizer input PDF satisfies the conditions for PQN, then for moment calculations, the quantizer may be replaced by a source of additive PQN.

But what about the correlation between the quantization noise and the quantizer input?

To answer this question, we will study the feedback system of Fig. 16.2(c). Here, the quantizer has been replaced by a source of additive independent noise n. The noise is PQN. Injection of noise n into the system causes noise to circulate throughout. The result is that noise present at the output of gain a is correlated with noise n. This is the same correlation that exists between the output of gain a and the actual quantization noise ν that exists in Fig. 16.2(a), assuming that QT II is satisfied at the quantizer input.

16.3 PQN MODELING AND MOMENTS WITH FIR AND IIR SYSTEMS

In Fig. 16.2(c), additive independent PQN noise is correlated with the output of gain a. When QT II is satisfied, the quantization noise in Fig. 16.2(a) is correlated in the same way with the output of gain a. The quantization noise and the additive PQN noise have exactly the same moments. The PQN model applies to the quantizer and the PQN model replaces the quantizer with a source of additive PQN noise, even though the quantization noise is correlated with the quantizer input.

If the multidimensional PDF of the quantizer input satisfies the conditions for PQN, then

(a) in an open loop system (FIR), the quantization noise will be uncorrelated with the quantizer input,

(b) in a closed loop feedback system (IIR), the quantization noise will be correlated with the quantizer input in the same way that it would be if the quantizer were replaced by a source of independent additive PQN.

If one were to test a system containing a quantizer in a feedback loop to see if conditions for PQN are met by the input to the quantizer, one would first check the quantization noise itself to make sure that it has zero mean, is uniformly distributed, and is white. The quantization noise would be obtained by subtracting the quantizer input from its output. One could further check to determine if the noise is uncorrelated over time. The crosscorrelation function between the quantizer input and the quantization noise would not be zero for all lags, but should be the same as if the quantizer were replaced with additive PQN. This crosscorrelation function should be zero for zero lag because there is a delay around the loop of at least one sample time, and should be other than zero for lags greater than zero. More will be said about this below.

16.4 ROUNDOFF IN A ONE-POLE DIGITAL FILTER WITH FIXED-POINT COMPUTATION

In order to implement the difference equation Eq. (16.1) or equivalently the block diagram of Fig. 16.1, certain roundoff operations will be necessary. The block diagram of Fig. 16.1 has been redrawn as Fig. 16.3, including the various quantization operations that we will soon see are necessary to perform the arithmetic.

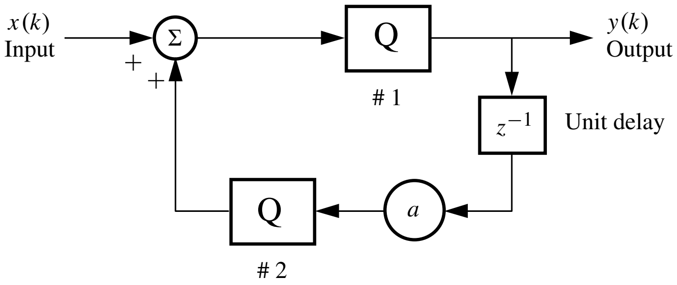

Figure 16.3 A one-pole digital filter with quantizers that are required for the arithmetic functions.

Regarding Fig. 16.3, a quantizer is placed in the feedback path after multiplication by the coefficient a. This is quantizer #2. As mentioned above, its purpose is to prevent the word length of the feedback signal from growing indefinitely.

The quantized feedback signal is summed with the input signal. If the input signal is quantized with the same granularity as that of the feedback signal, addition could take place with no further quantization. But if the input is much more finely quantized, then the addition could be performed accurately, but the result would need to be quantized to assure that it conforms to the number scale within the feedback path. Let us assume that this is the case, and we will include quantizer #1 in our example having the same quantization step size as quantizer #2.

Thus, there are two quantizers within the feedback loop. Since the input $x(k)$ is finely quantized, the summed signal applied to quantizer #1 will satisfy conditions for PQN to a close approximation. Furthermore, the input to quantizer #2 will dynamically vary and in most cases will approximately satisfy the conditions for PQN for quantizer #2. The two quantization noises will be highly uncorrelated with each other, and will essentially be white. But there is no guarantee that this will be true in all cases.

Replacing the quantizers with additive white independent noises as shown in Fig. 16.4, the filter output noise power can be readily estimated. The output noise power from quantizer #1 is obtained in the following way. We first find the transfer function from the point of injection of this noise to the filter output $y(k)$. This transfer function is

$$\begin{pmatrix}\text{transfer function}\\ \text{from quantizer \#1}\\ \text{to filter output}\end{pmatrix} = \frac{1}{1-az^{-1}}\,. \tag{16.3}$$

The corresponding impulse response, the inverse transform of Eq. (16.3), is the following geometric series:

$$\begin{pmatrix}\text{impulse response}\\ \text{from quantizer \#1}\\ \text{to filter output}\end{pmatrix} = 1, a, a^2, a^3, \dots\,. \tag{16.4}$$

The sum of the squares of the impulses of the impulse response is

$$\begin{pmatrix}\text{sum of}\\ \text{of}\\ \text{squares}\end{pmatrix} = 1 + a^2 + a^4 \cdots = \frac{1}{1-a^2}\,. \tag{16.5}$$

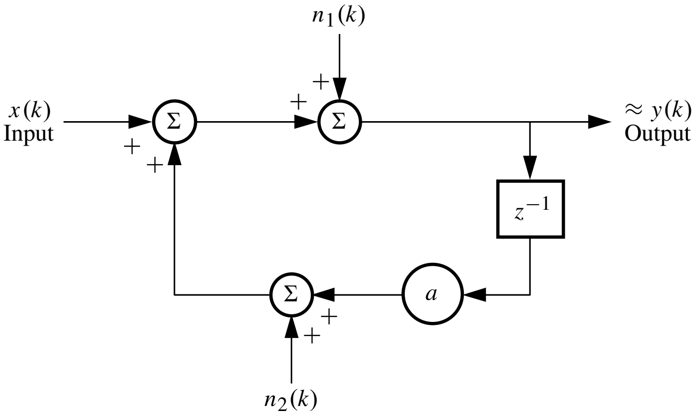

Figure 16.4 A one-pole digital filter with additive independent white noises replacing the quantizers.

Assuming that the quantization noise from quantizer #1 is white with zero mean and a mean square of $q^2/12$, the output noise power due to this quantizer will be

$$\frac{q^2}{12}\cdot\begin{pmatrix}\text{sum}\\ \text{of}\\ \text{squares}\end{pmatrix} = \frac{q^2/12}{1-a^2}\,. \tag{16.6}$$

Assuming that the quantization noise from quantizer #2 has the same properties as that of quantizer #1 and that both noises are uncorrelated with each other, the output noise power due to quantizer #2 will also be

$$\frac{q^2}{12} \cdot \begin{pmatrix} \text{sum} \\ \text{of} \\ \text{squares} \end{pmatrix} = \frac{q^2/12}{1-a^2}\,. \tag{16.7}$$

Note that the impulse response from quantizer #2 to the filter output is the same as that from quantizer #1 to the filter output. The total output noise power is the sum of the noise powers,

$$\begin{pmatrix} \text{total} \\ \text{output} \\ \text{quantization} \\ \text{noise power} \end{pmatrix} = \frac{q^2/6}{1-a^2}\,. \tag{16.8}$$

Suppose, for example, that the input signal is a sampled low-frequency sine wave, i.e. samples of $A \sin \omega t$. The low-frequency gain of the filter is

$$\begin{pmatrix} \text{input/output} \\ \text{transfer} \\ \text{function} \end{pmatrix}_{z^{-1}\approx 1} = \left(\frac{1}{1-az^{-1}} \right)_{z^{-1}\approx 1} \approx \frac{1}{1-a}\,. \tag{16.9}$$

The output will therefore be samples of the sine wave $\left(\frac{A}{1-a}\right) \sin \omega t$. The signal power at the output will be

$$\begin{pmatrix} \text{output} \\ \text{signal} \\ \text{power} \end{pmatrix} = \frac{1}{2} \left(\frac{A}{1-a} \right)^2\,. \tag{16.10}$$

Let the amplitude A be chosen to exercise the quantizers at approximately full-scale level, but not to overload them. Regarding Fig. 16.3, the signal level at quantizer #1 is the same as that at the filter output, and the signal level at quantizer #2 is a times this. The magnitude of a must be less than 1 for the filter to be stable. So keeping the signal level under the saturation limit for quantizer #1 is appropriate.

Suppose for sake of example that both quantizers are 12-bit quantizers. The range of the quantizers would therefore be $\pm q \cdot 2^{11}$. Then the maximum output signal magnitude would be

$$\frac{A}{1-a} = q \cdot 2^{11}\,. \tag{16.11}$$

The output signal-to-noise ratio can be calculated in the following way. The output noise power is given by Eq. (16.8), the output signal power is given by Eq. (16.10), and a relation between A and q is Eq. (16.11). Accordingly,

$$\begin{aligned}\begin{pmatrix}\text{output}\\ \text{SNR}\end{pmatrix} &= \frac{\frac{1}{2}(\frac{A}{1-a})^2}{(\frac{q^2/6}{1-a^2})}\\ &= \frac{\frac{1}{2}q^2 2^{22}}{(\frac{q^2/6}{1-a^2})}\\ &= (1-a^2)3\cdot 2^{22}\\ &= 1.26\cdot 10^7(1-a^2)\\ &= 71.0 + 10\log_{10}(1-a^2)\,\text{dB}\,. \end{aligned} \qquad (16.12)$$

The output SNR will be 71 dB or less, depending on the choice of a.

Assuming that the PQN model applies for both quantizers, one could compute the output SNR for quantizers with various numbers of bits, for sinusoidal inputs that may not be full-scale for the quantizers, and whose frequencies may be other than "low" so that the gain of the filter would be different from that given by Eq. (16.9).

Example 16.1 Quantization Noise with Sinusoidal Input

Let us consider the system given in Fig. 16.3, with 8-bit quantizers. The output response to a sine wave input turned on at time zero is shown in Fig. 16.5(a). The same system was run with "infinitely" fine quantization, and the difference between the system with 8-bit quantization and the "perfect" system is shown in

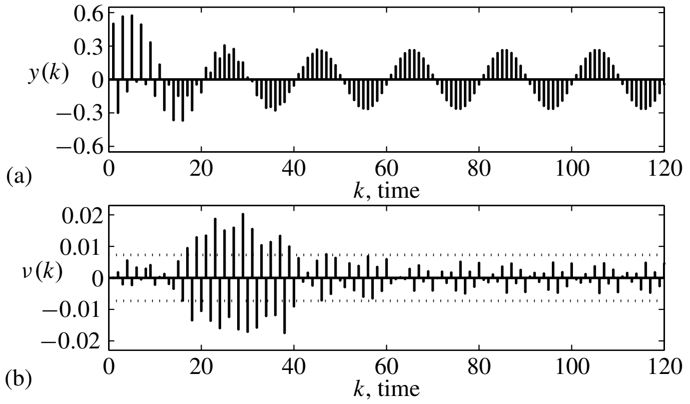

Figure 16.5 Response to a sine wave of frequency (sampling frequency)/20 applied to the system of Fig. 16.3, turned on at time $k = 0$, with 8-bit quantizers working in the amplitude range $[-1, 1]$, with the parameter $a = 0.11101$ in binary, approximately equal to 0.906, and an initial value $y(0) = 0$: (a) output response; (b) output quantization noise, with theoretical standard deviations marked with dotted lines.

Fig. 16.5(b). This difference is the output quantization noise whose variance was theoretically predicted using the PQN model. The theoretical standard deviations are indicated by the dotted lines. In steady state, after the initial transient dies out, the output quantization noise is well within ± one standard deviation, indicating that quantization noise of the quantizers in the feedback link is less than that predicted by PQN.

It would also be possible to predict output SNR if the filter input were stochastic, let us say Gaussian. The methodology would be similar.

Example 16.2 Quantization Noise with Gaussian Input
Let us consider the system given in Fig. 16.3, with 8-bit quantizers. The response to a Gaussian input turned on at time zero, and the time function of the output quantization noise are shown in Fig. 16.6(a)–(b), respectively. The power of the output quantization noise comes close to that predicted theoretically.

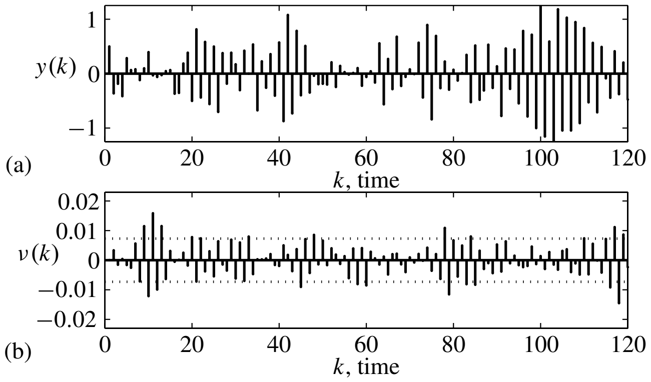

Figure 16.6 Response to white zero-mean Gaussian input, with $\sigma = 0.25$, applied to the system of Fig. 16.3, turned on at time $k = 0$, with 8-bit quantizers working in the amplitude range $[-1, 1]$, with parameter $a = 0.11101$ in binary, approximately equal to 0.906, and an initial value $y(0) = 0$: (a) output response; (b) output quantization noise, with theoretical standard deviations marked with dotted lines.

The big question is, does the PQN model apply to the quantizers? Regarding Fig. 16.3 once again, it is clear that input conditions for quantizer #1 will generally support this model, since its input is a mixture of the filter input and the feedback signal. The filter input has a smooth PDF, and when it mixes with the feedback which has a discrete PDF, the result is generally a smooth PDF. If the filter input and the feedback signal were statistically independent, then the input PDF to quantizer #1 would be the convolution of the two PDFs, and the smoothing effect would be clear. The two added signals are generally not statistically independent, but are correlated, so the two PDFs do not convolve, but the PDF of the sum is still generally smooth.

However, the PDF of the input to quantizer #2 is discrete, not smooth. The effect that this has on quantizer #2 is determined by the value of a. If the value of a is near to 1, then quantizer #2 may not be "doing much quantization." Its inputs would almost always be in the center of its quantum boxes. That being the case, the noise injected by this quantizer may not be uniformly distributed and may not have the value $q^2/12$. For other values of a, it is even possible that the successive quantization error samples always have opposite signs or have the same sign for several cycles, forcing the loop to continuously oscillate. The following example illustrates this case.

Example 16.3 Limit-Cycle Oscillation
Let us consider the one-pole system given in Fig. 16.3, with 8-bit quantizers. The response is shown in Fig. 16.7, with parameters described in the caption of the figure. Although the input of the system is a constant, turned on at time $k = 0$, its output yields an oscillatory pattern called a "limit cycle." The amplitude of the oscillation in $y(k)$ is about ±0.031, 8 times larger than the maximum quantization error which is only $q/2 = 0.0039$, and about 4 times larger than the theoretical standard deviation of the output noise, obtained from Eq. (16.8) as 0.0073. Limit cycles occur with this system only under very special conditions of input and initial conditions, but their occurrence could be highly undesirable.

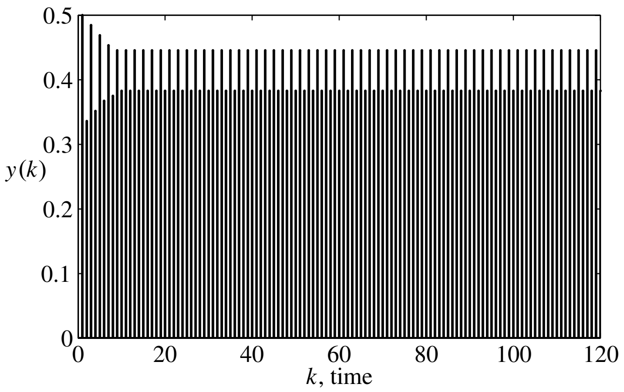

Figure 16.7 Limit cycle oscillation at the output of the system of Fig. 16.3, with 8-bit quantizers working in the amplitude range $[-1, 1]$, with parameter $a = 0.11101$ in binary, approximately equal to 0.906, a constant input of $x(k) = 0.1100101$ in binary, approximately equal to 0.789, and an initial value $y(0) = 0.1$ in binary which is equal to 0.5.

This unwanted phenomenon can be eliminated by the use of additive dither (see details later in Chapter 19). We can add e.g. independent dithers, uniformly distributed between $\pm q/2$, to the input signals of both quantizers as shown in Fig. 16.8. The dither linearizes the quantizers. Without quantization, the system is linear, and no limit cycle phenomena would occur. Fig. 16.9 shows the output

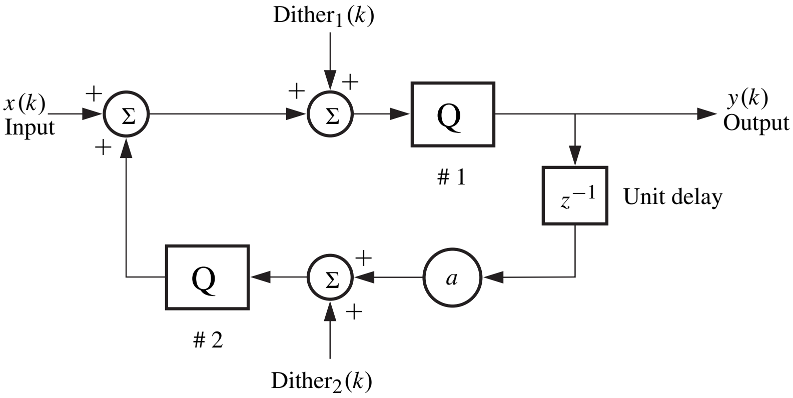

Figure 16.8 The system of Fig. 16.3 with random dither signals added to the quantizer inputs

of this same system without quantization (with "infinitely fine" quantization), all other parameters being the same. The input is a constant, turned on at time $k = 0$. The output of the system with 8-bit quantization and with dither applied

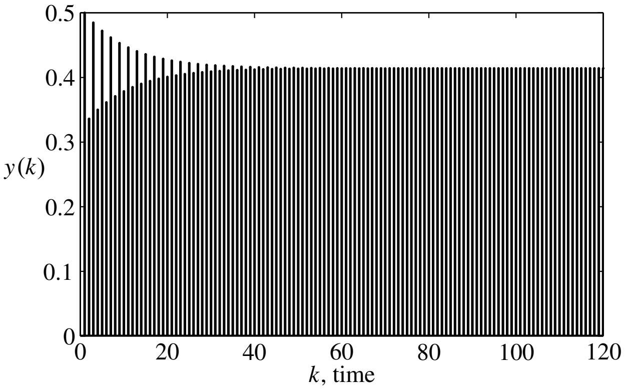

Figure 16.9 Response of the "infinitely finely" quantized system (simulated with IEEE double precision) to the same input as in Fig. 16.7.

to both quantizers is shown in Fig. 16.10. The limit cycles seen in Fig. 16.7 have been stopped.

The question remains, does the PQN model apply to the quantizers? The answer is not easy to get analytically for a feedback system in general cases. More will be said below about using computational techniques to test for PQN, and this will be discussed further in Chapter 17 on feedback control systems. Our experience has been that the PQN model works amazingly well in very large numbers of cases, but care

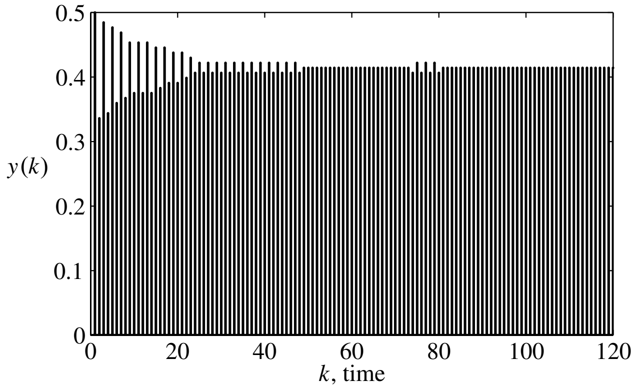

Figure 16.10 Output response of the quantized system with dither. The limit cycle oscillation is stopped in the system of Fig. 16.3, with the same settings as in Fig. 16.7, by adding independent dither, uniformly distributed between $\pm q/2$, to both quantizer inputs. This output response is a bit noisy since it contains quantization noise and components of the dither.

must be taken especially with multiple quantizers (multiple arithmetic operations executed, especially when using low bit numbers) in a feedback loop. This is a subject that warrants further research.

16.5 ROUNDOFF IN A ONE-POLE DIGITAL FILTER WITH FLOATING-POINT COMPUTATION

When implementing the one-pole filter of Fig. 16.1 with floating-point arithmetic, roundoff will be present for the same reasons as existed with fixed-point integer arithmetic. Therefore, the feedback diagram representing the actual implementation will contain quantizers located as before, and as diagrammed in Fig. 16.3. The only difference is that the quantizers are floating-point quantizers. The analysis of noise in the filter output due to quantization will be similar to the previous analysis, differing primarily in the amount of quantization noise induced into the system by each quantizer.

The first issue to be addressed is about the application of floating-point PQN modeling to the two quantizers. The input to quantizer #1 more easily satisfies the conditions for PQN. The input to quantizer #2 is quantized, and satisfaction of QT II for this quantizer will never happen perfectly. Approximate application of the PQN model is possible, and will depend on signal characteristics and choice of the parameter a. Following the argument of Section 13.7 however, the applicability of the PQN model depends not so much on quantizer input PDF as on the length of the quantizer's mantissa. With a mantissa of 16 bits or more, the PQN model almost always works well.

It is useful to determine the output signal-to-quantization noise ratio. We assume that the PQN model applies to both quantizers, and proceed to determine output SNR. For moment calculations, quantizers are regarded as additive independent noise sources, as in the diagram of Fig. 16.4.

We need to obtain the quantization noise power injected by quantizer #1. The power of the input to this quantizer is approximately equal to the power of the filter output $y(k)$. Therefore, the noise power injected by this quantizer is

$$\begin{pmatrix}\text{noise power of}\\ \text{quantizer \#1}\end{pmatrix} = 0.180 \cdot 2^{-2p} \cdot \mathrm{E}\{y^2(k)\}\,. \tag{16.13}$$

The noise power at the filter output due to white noise injected by quantizer #1 is obtained by multiplying Eq. (16.13) by the sum of squares of the impulse response from quantizer to output as given by Eq. (16.5). The result is

$$\begin{pmatrix}\text{output noise}\\ \text{power due to}\\ \text{quantizer \#1}\end{pmatrix} = \frac{0.180 \cdot 2^{-2p} \cdot \mathrm{E}\{y^2(k)\}}{1-a^2}\,. \tag{16.14}$$

Next we will calculate the output noise power due to quantizer #2. The impulse response from the point of injection of the quantization noise to the filter output is the same as Eq. (16.4). The injected noise power is different from that of quantizer #1, however. Inspection of Fig. 16.4 allows us to deduce that the input power to quantizer #2 is equal to the filter output power multiplied by a^2. Therefore the quantization noise power injected by this quantizer is

$$\begin{pmatrix}\text{noise power}\\ \text{of quantizer \#2}\end{pmatrix} = 0.180 \cdot 2^{-2p} a^2 \mathrm{E}\{y^2(k)\}\,. \tag{16.15}$$

The noise power at the filter output due to white noise injected by quantizer #2 is

$$\begin{pmatrix}\text{output noise}\\ \text{power due to}\\ \text{quantizer \#2}\end{pmatrix} = \frac{0.180 \cdot 2^{-2p} a^2 \mathrm{E}\{y^2(k)\}}{1-a^2}\,. \tag{16.16}$$

The total output quantization noise power is the sum of Eq. (16.16) and Eq. (16.14).

$$\begin{pmatrix}\text{total output}\\ \text{quantization noise}\end{pmatrix} = 0.180 \cdot 2^{-2p} \mathrm{E}\{y^2(k)\} \left(\frac{1+a^2}{1-a^2}\right)\,. \tag{16.17}$$

The output signal-to-quantization-noise ratio is

$$\mathrm{SNR} = \frac{5.55(1-a^2)2^{2p}}{1+a^2}\,. \tag{16.18}$$

Expressed in decibels, this is

$$\text{SNR, dB} = 10\log_{10}\left(\frac{5.55(1-a^2)2^{2p}}{1+a^2}\right). \qquad (16.19)$$

This formula will work for sinusoidal or Gaussian input signals or for any input that allows the application of the PQN model to the quantizers.

The digital filter will be stable as long as

$$|a| < 1\,. \qquad (16.20)$$

If we choose a to be 0.75 for example and use IEEE single-precision floating-point computation, the SNR will be

$$\text{SNR, dB} = 10\log_{10}\left(\frac{5.55(1-0.5625)2^{48}}{1+0.5625}\right) = 146\,\text{dB}\,. \qquad (16.21)$$

The effects of quantization noise will be very small indeed for this case.

If a is less than 1 but very close to it, the digital filter will be almost unstable. The impulse response (16.4) will decay very slowly, and there will be significant accumulation of roundoff noise in the feedback loop. Expressions Eq. (16.18) and Eq. (16.19) show that SNR will be much smaller when a is close to 1. Suppose that a has the value 0.9999. The SNR will now be

$$\text{SNR, dB} = 10\log_{10}\left(\frac{5.55(1-0.9998)2^{48}}{1+0.9998}\right) = 112\,\text{dB}\,. \qquad (16.22)$$

The SNR is a lot lower, but it would still be an excellent SNR for most applications.

Notice that the SNR is not a function of the signal level, as is generally the case with floating-point computation.

16.6 SIMULATION OF FLOATING-POINT IIR DIGITAL FILTERS

Quantization noise in digital filters can be studied in simulation by comparing the behavior of the actual quantized digital filter with that of a reference digital filter having the same structure but whose numerical calculations are done extremely accurately. Suppose that the filter of Fig. 16.3 is to be implemented with single-precision arithmetic, and that the reference filter is implemented with double-precision arithmetic. Great care must be taken to insure that the feedback coefficient a is identical for both filters. This can be done for both filters by using the single-precision version of a as this coefficient would be implemented in the single-precision filter.

Fig. 16.11 is a block diagram showing both filters. The reference filter performs arithmetic essentially perfectly, so it is shown without quantization.

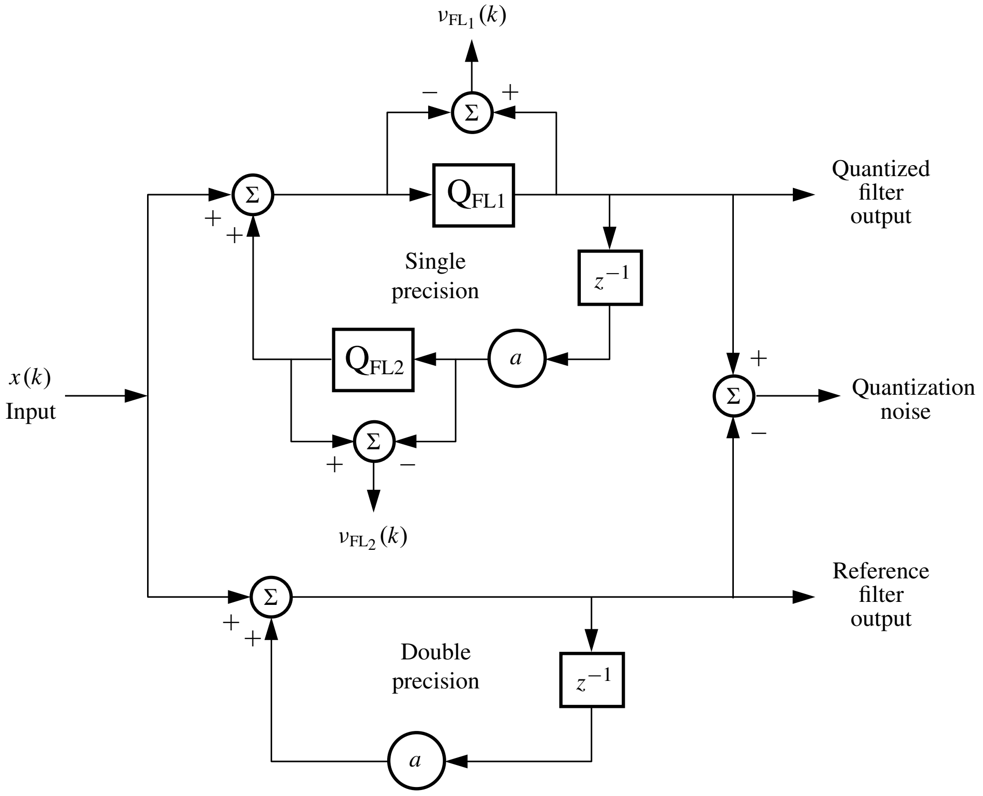

Figure 16.11 Comparison of quantized digital filter (single precision, that is, the precision is $p = 24$) with "perfect" reference filter (double precision, that is, $p = 53$).

The difference between the outputs of the two filters in Fig. 16.11 is a very close approximation to the total quantization noise at the output of the quantized filter. Simulation experiments have been done to measure the quantization noise power, and to measure its correlation with the output of the reference filter. Correlations between the quantization noises of the two quantizers have been measured, as well as have correlations between the quantization noises and the respective quantizer input signals, and the powers of the two quantization noises have been measured. Results come out exactly as predicted in the previous section when using the PQN model for the quantizers.

Choosing a feedback coefficient of $a = 0.75$, the signal-to-noise ratio for the quantized filter was measured by measuring the power of the pure underlying signal, i.e. the power of the output of the reference filter, and dividing this by the output quantization noise power. With a white Gaussian input $x(k)$, the measured SNR was

$$\mathrm{SNR} = 146\,\mathrm{dB}\,. \tag{16.23}$$

With a sinusoidal input $x(k)$, the measured SNR was

$$\mathrm{SNR} = 146\,\mathrm{dB}\,. \tag{16.24}$$

These measurements compare almost perfectly with the theoretical SNR given by Eq. (16.21) for single-precision arithmetic.

Now choosing a feedback coefficient of $a = 0.9999$, further measurements can be made. This filter has a very long time constant, and the sum of squares of the impulses of the impulse response is much larger than before. The same kinds of measurements have been made for this case.

With a white Gaussian input $x(k)$, the measured SNR was

$$\mathrm{SNR} = 118\,\mathrm{dB}\,. \tag{16.25}$$

With a sinusoidal input $x(k)$, the measured SNR was

$$\mathrm{SNR} = 113\,\mathrm{dB}\,. \tag{16.26}$$

These measurements compare well with the theoretical results of Eq. (16.22).

16.7 STRANGE CASES: EXCEPTIONS TO PQN BEHAVIOR IN DIGITAL FILTERS WITH FLOATING-POINT COMPUTATION

With a mantissa length of 16 bits or more, the PQN model almost always works extremely well, in every respect, for the two quantizers of Fig. 16.3. For moment calculations, the quantizers may be replaced by sources of additive independent white noises.

There are strange cases that may arise however where the PQN model breaks down. These cases result from the fact that the input PDF of quantizer #2 is discrete, that is, in this quantizer, re-quantization happens. For strange cases, the quantizing theorem is not at all satisfied for quantizer #2. The strange cases occur for rare choices of the feedback coefficient a. Almost any choice of the coefficient will result in the PQN model working well. But there are exceptions, and some of them will be described next.

Here is a strange case: Let $a = 1/2$. Now we find that quantizer #2 produces no quantization noise, while quantizer #1 produces noise in accord with the PQN model. So the output noise is less by the amount of noise that would have come from quantizer #2. We have the same effect for $a = 1/4$, or $1/8$, or $1/16$, and so on. The reason is that the amplitude of each input sample to quantizer #2 occurs exactly at the center of a floating-point quantization box. So quantizer #2 produces no noise. Knowing this, one could easily compute the output quantization noise power.

Other strange cases occur when a is just slightly greater or slightly less than $1/2$. These cases have not yet been investigated in detail. The whole subject of testing for applicability of the PQN model in closed-loop systems and the determination of output quantization noise when PQN does not apply is an important area of study that needs more work.

Computation techniques for PQN testing is a subject that will be treated below.

16.8 TESTING THE PQN MODEL FOR QUANTIZATION WITHIN FEEDBACK LOOPS

A series of measurements have been made with the filter of Fig. 16.2 with a white Gaussian input and with a sinusoidal input. The feedback coefficient was set at $a = 0.99$. The arithmetic was single-precision ($p = 24$) floating-point. The objective of these measurements was to make definitive tests of the behavior of the two quantizers to determine the applicability of the PQN model.

The block diagram of Fig. 16.11 was very useful for this testing. In addition, the PQN model of the system of Fig. 16.11 is shown in 16.12. For this PQN model to apply in a strict sense, all moments and joint moments must be correspondingly alike for the systems of Fig. 16.11 and Fig. 16.12. Of greatest interest in fact are the first- and second-order moments; i.e. mean squares, autocorrelation functions, and crosscorrelation functions.

Our testing for PQN focused on the first- and second-order moments of the systems of Fig. 16.11 and Fig. 16.12 with a white Gaussian input and with a sinusoidal input. In practice, this kind of testing should be done with the actual filter input signal.

The first test that was performed measured the output SNR of Fig. 16.11 to see if this would be the same as the output SNR in Fig. 16.12. For the PQN model, the theoretical output SNR is given by Eq. (16.19). Applying this to the present case,

$$\begin{aligned} \mathrm{SNR} &= 10\log_{10}\left(\frac{5.55(1-a^2)2^{2p}}{1+a^2}\right) \\ &= 10\log_{10}\left(\frac{5.55\cdot 0.02\cdot 2^{48}}{2}\right) \\ &= 132\,\mathrm{dB}\,. \end{aligned} \tag{16.27}$$

The actual measurements showed that for the quantized filter, the output SNR $=$ 131.1 dB with Gaussian input and SNR $=$ 132.6 dB with sinusoidal input. For the PQN model, the theoretical output SNR $=$ 132 dB with both Gaussian or sinusoidal input.

Next, the crosscorrelation function was obtained by measurement between the output quantization noise and the underlying output signal, the reference filter output in Fig. 16.11. The result is shown in Fig. 16.13. This was compared with the crosscorrelation function between the output PQN noise and the underlying output signal, the reference filter output in Fig. 16.12. Theoretically this crosscorrelation function should be zero for the PQN system because the noise is caused by injection of two

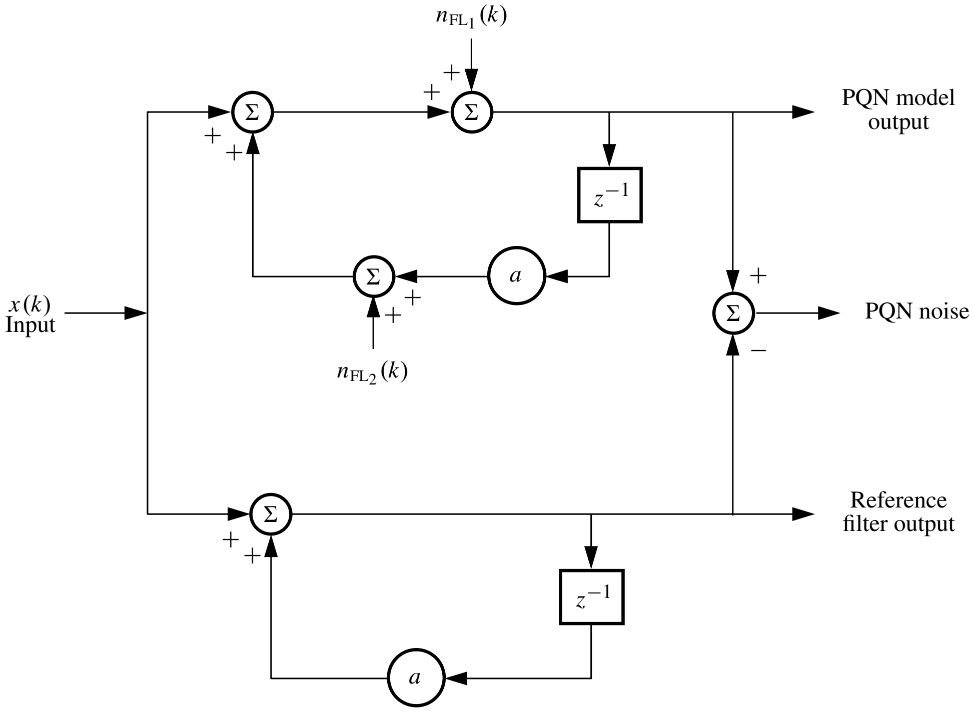

Figure 16.12 Testing the PQN model of a quantized digital filter.

independent PQN sources, which are independent of the input $x(k)$. The measured crosscorrelation function of Fig. 16.14 shows this within experimental errors. The quantization noises on the other hand are deterministically related to this input signal, but, nevertheless, the crosscorrelation function between the output quantization noise and the underlying output signal is zero for all lags, within experimental error, according to the plot of Fig. 16.13. So far, the PQN model passes this test.

Next, the autocorrelation functions of the output quantization noise in Fig. 16.11 were measured, and they are shown in Fig. 16.15(a) for a Gaussian input and in Fig. 16.15(b) for a sinusoidal input. Corresponding autocorrelation functions for the output PQN noise (Fig. 16.12) were measured, and they are shown in Fig. 16.16(a) for the Gaussian input and in Fig. 16.16(b) for the sinusoidal input. Within experimental error, both functions are the same. For the PQN case, the two white noises go through the same transfer function $(\frac{1}{1-az^{-1}})$ to reach the filter output. The z-transform of the autocorrelation function should be proportional to

$$\left(\frac{1}{1-az^{-1}}\right)\left(\frac{1}{1-az}\right). \tag{16.28}$$

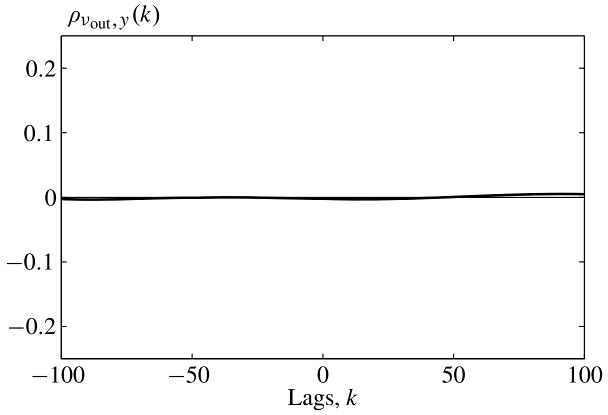

Figure 16.13 Measured correlation coefficient between the output quantization noise and the underlying output signal, as a function of the lags (number of sample periods). $a = 0.99$, $p = 24$, white Gaussian input with zero mean and variance 1.

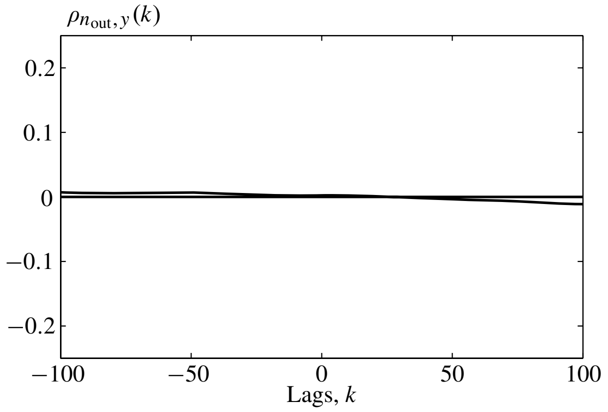

Figure 16.14 Measured correlation coefficient between the output noise due to PQN and the underlying output signal, as a function of the lags (number of sample periods). $a = 0.99$, $p = 24$, white Gaussian input with zero mean and variance 1.

This corresponds to a two-sided symmetrical exponential autocorrelation function, as verified by the experimental plots of Figs. 16.16 and 16.15.

The quantization noises ν_{FL_1} and ν_{FL_2} (see Fig. 16.11) were checked, and their means were zero within experimental error. The mean square values of these noises were measured and they agreed with PQN theory, all within experimental error.

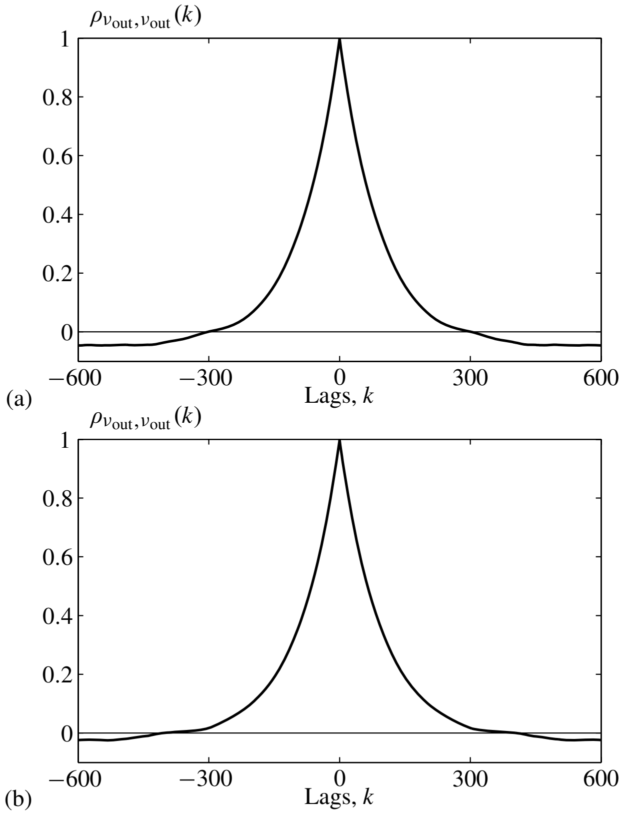

Figure 16.15 Measured autocorrelation functions of output quantization noises: (a) for white Gaussian input with zero mean and variance 1; (b) for sinusoidal input, $T_{\text{per}} \approx$ 628 lags, $A = \sqrt{2}$.

When crosscorrelating these noises, their crosscorrelation function was zero for all lags, within experimental error.

In an open-loop system, when QT II is satisfied by a quantizer input signal and the PQN model applies, the quantization noise is uncorrelated with the quantizer input signal. This is not the case in a feedback system, and measurements were made to verify this.

Regarding Fig. 16.11, crosscorrelation functions between the quantization noises and the quantizer input signals were measured. Regarding Fig. 16.12, corresponding crosscorrelation functions between the PQN noises and the inputs to the PQN sources were measured by running the system of Fig. 16.17b. The quantization noise signals ν_{FL_1} and ν_{FL_2} were taken from the system of Fig. 16.11.

The crosscorrelations between the injected noises and the input signals at the noise injection points in Fig. 16.17 were measured. The transfer functions from

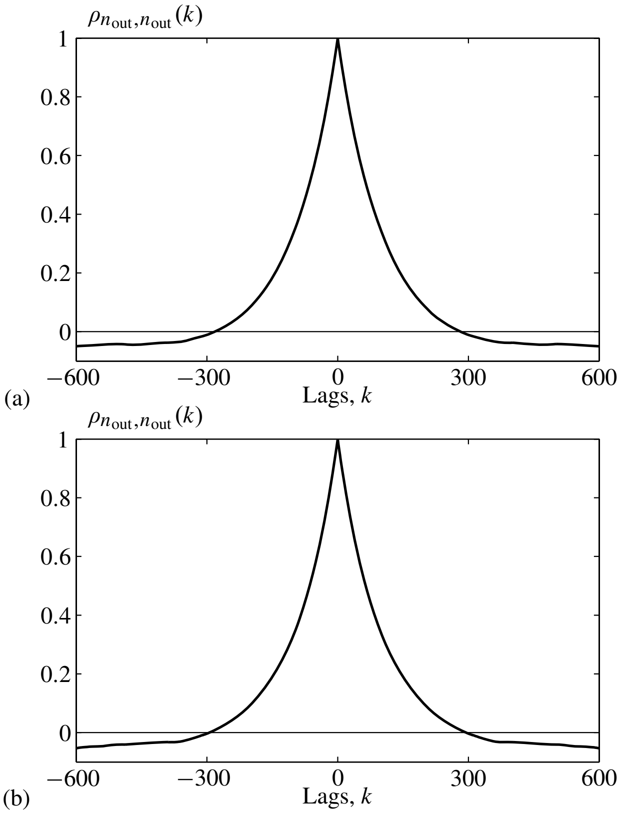

Figure 16.16 Measured autocorrelation functions of the output noise when the PQN model applies: (a) for white Gaussian input with zero mean and variance 1; (b) for sinusoidal input, $T_{\text{per}} \approx 628$ lags, $A = \sqrt{2}$.

injection noise point around the loop back to this point were all the same, i.e. $\frac{1}{1-az^{-1}}$. Therefore, the z-transform of the crosscorrelation functions for the PQN case should be proportional to $\frac{1}{1-az^{-1}}$, and the crosscorrelation functions should be exponential steps. The measured crosscorrelation function for the #2 PQN source is shown in Fig. 16.18.

The measured crosscorrelation function for the #2 quantization noise source is shown in Fig. 16.19(a) with Gaussian input and in Fig. 16.19(b) with sinusoidal input.

The correlation functions of Fig. 16.19 all correspond to the correlation functions of the PQN case as shown in Fig. 16.18. Similar correspondence was obtained with the #1 PQN source and the #1 quantization noise source. The correspondence in these correlation functions is further evidence of the validity of the PQN model for the two quantizers of Fig. 16.3.

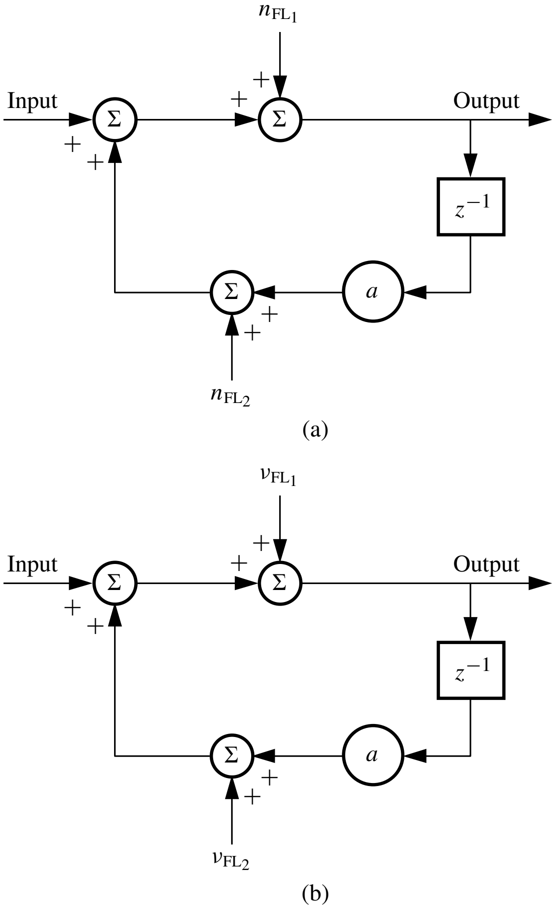

Figure 16.17 Measurement of crosscorrelation between injected noises and input signals at the noise injection points: (a) the PQN model; (b) actual quantization noise.

Every test that we made showed that the moments and joint moments of the quantized filter were the same when the quantizers were replaced with additive PQN. This is very nice. However, in a practical application, more extensive tests are necessary to detect unlikely but possible strange cases. Moreover, the applicability of PQN can be assured by the application of dither within the loop (see Chapter 19).

Software that allows one to do PQN testing for the quantizers in any digital system is available from the home page of this book.[1]

[1] `http://www.mit.bme.hu/books/quantization/`

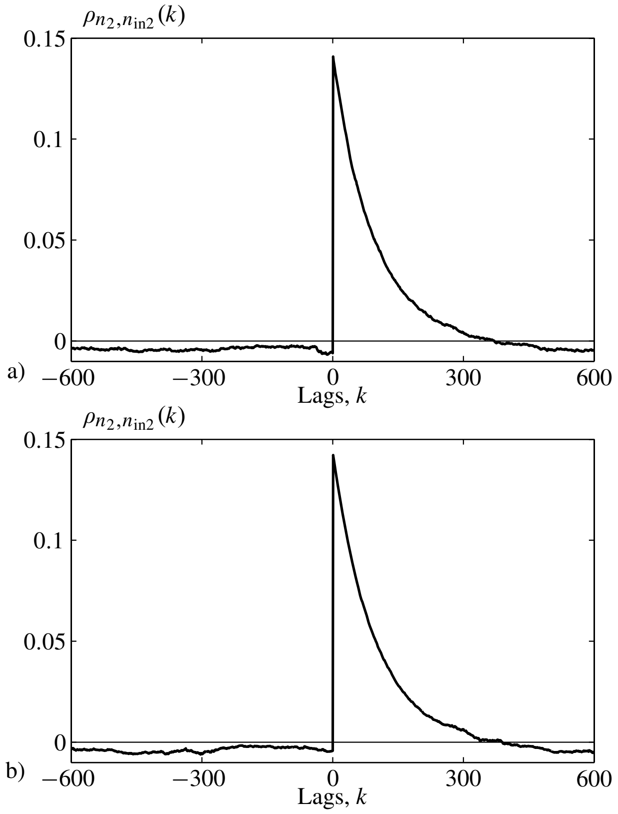

Figure 16.18 Measured crosscorrelation functions between injected PQN noise for quantizer #2 and PQN noise at its input: (a) for white Gaussian input with zero mean and variance 1; (b) for sinusoidal input, $T_{\text{per}} \approx 628$ lags, $A = \sqrt{2}$.

16.9 SUMMARY

Quantization takes place within the feedback loops of IIR digital filters. When conditions at the inputs of the quantizers satisfy QT II, and this is most frequently the case for both fixed-point and floating-point arithmetic, the quantizers may be replaced by sources of additive independent PQN.

The mean square of the quantization noise at the output of a digital filter equals the sum of the mean squares of the noise components at the filter output. The mean square of each component of the output noise may be computed. This may be done by multiplying the mean square of the quantization noise at the quantizer by the sum of squares of the samples of the impulse response of the path from the quantizer to the output of the digital filter. Sample computations are given for a one-pole filter.

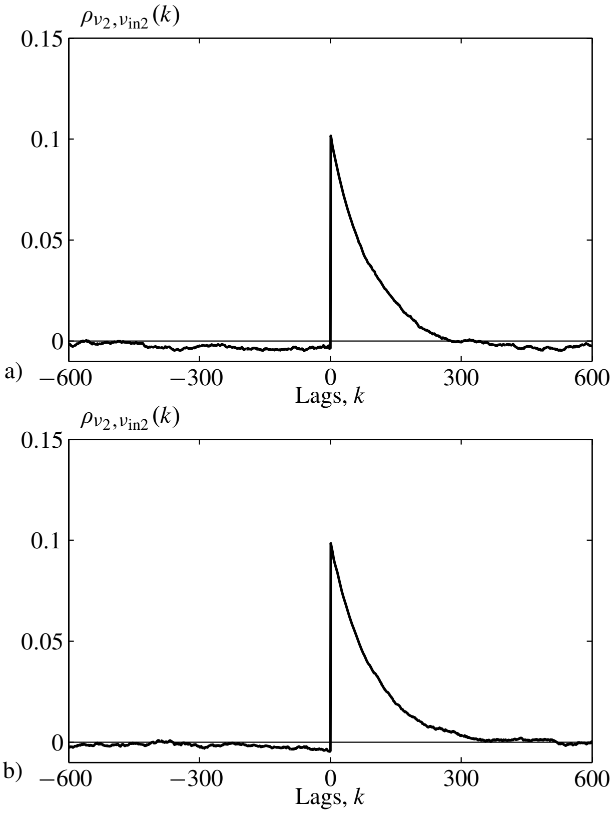

Figure 16.19 Measured crosscorrelation functions between generated quantization noise and input quantization noise for quantizer #2: (a) for white Gaussian input with zero mean and variance 1; (b) for sinusoidal input, $T_{\text{per}} \approx 628$ lags, $A = \sqrt{2}$.

Verification of the PQN model is demonstrated by comparing various auto- and crosscorrelation functions measured when sinusoidal and Gaussian inputs were applied to the digital filter that was simulated with quantization, and without quantization but with appropriate injection of PQN noise. There was complete correspondence of moments.

Chapter 17 extends the analysis of this chapter to more complex feedback systems. One of the subjects to be discussed in that chapter is the Bertram bound that allows one to bound the output quantization noise in feedback systems. It is useful to compare the output noise standard deviation with the bound. This will be done in Chapter 17.

16.10 EXERCISES

16.1 Consider the system depicted in Fig. 16.1 (page 403), with $a = 0.625$. Let the number representation be 8-bit two's complement fixed-point (see page 1.6a) with the binary point after the sign bit: e.g. 01000000 represents the digital number 0.5, and 10100000 represents -0.75. Thus, 0.625 is represented as 0.1010000, a number with two bits only with value 1. Apply a zero mean white Gaussian input with $\sigma = 0.01$.

- (**a**) Determine the quantization noise variance at the output, using PQN to model arithmetic roundoff.
- (**b**) Verify using Monte Carlo simulation if the PQN model is really applicable. If there is deviation, how large is it?

16.2 A nonzero-mean Gaussian signal of bandwidth $B = 2\,\text{kHz}$ is recursively averaged using the following algorithm: $\hat{\mu}(k+1) = \hat{\mu}(k) + \frac{1}{C}(x(k) - \hat{\mu}(k))$, $k = 0, 1, ...,$ $\hat{\mu}(0) = 0$. The value of C is 50, the sampling frequency is $f = 4\,\text{kHz}$.

- (**a**) How many steps are necessary to approximately reach the steady state? How large is the measurement time?
- (**b**) How large is the reduction factor of the variance in steady state?
- (**c**) What happens if the sampling frequency is increased by a factor of 4 and the algorithm remains unchanged?
- (**d**) What happens if the sampling frequency is decreased by a factor of 4?
- (**e**) What is the effect if we change the initial value for $\hat{\mu}(0) = x(0)$?
- (**f**) Determine the value of C as a function of the sampling frequency f, if the equivalent bandwidth (B) and the desired decrease of the variance ($1/M$) are given.

Hint: for uncorrelated input samples of variance σ^2, the variance of the output approximately equals $\sigma^2/(2C)$.

16.3 We would like to measure the mean value of a bandlimited white noise signal with bandlimit $B = 10\,\text{kHz}$. We can take samples at a rate of $f_\text{s} = 10\,\text{kHz}$, and average the samples according to the following algorithm: $\hat{\mu}(k+1) = 0.95\hat{\mu}(k) + 0.05x(k)$. Determine the variance of the estimated mean for $k \gg 1$, if $\sigma_x^2 = 10$.
Hint: see the hint of Exercise 16.2.

16.4 Consider the system given in Example 16.3 (page 412).

- (**a**) Determine the total output noise power, based on PQN.
- (**b**) Calculate the power of the oscillation shown in the example. Which is larger? Why?
- (**c**) Determine by simulation what happens when independent, white, uniformly distributed between $-q/2$ and $q/2$ dither is added to the input of each quantizer before quantization.

16.5 Moving averaging can be re-formulated in recursive form:

$$y(k) = \frac{1}{N}\sum_{m=k-N}^{k-1} x(m) = y(k-1) + \frac{1}{N}\big(x(k-1) - x(k-N-1)\big)\,. \quad \text{(E16.5.1)}$$

Theoretically this has finite impulse response, but as implemented in a recursive form, it is not stable.

(a) Show that Eq. (E16.5.1) is not stable: there is a pole at $z = 1$.

(b) Why does instability occur in the recursive implementation, and not in the FIR implementation?

(c) What is the consequence of this instability to roundoff?

(d) Does the PQN model show this?

(e) What is the effect of changing the feedback factor from $a = 1$ to $a = 0.99$?

(f) Can the dc transfer function be compensated by changing also the feedforward coefficient -1 to a smaller value? Can this network be used to measure a dc value?

(g) How large is the noise in the output of the modified network on the basis of PQN?

(h) Verify the results of the previous item by numerical evaluation, and Monte Carlo simulation, assuming that $y(k) \approx 0.5$,

i. for fixed-point with $B = 16$ bits, two's complement in $(-1, 1)$.

ii. for floating-point with $p = 12$ bits.

16.6 Design direct implementation (evaluate $y(k) = 0.6y(k-1) - 0.05y(k-2) + 0.4x(k)$) of the second-order IIR filter

$$H(z) = \frac{0.4}{1 - 0.6z^{-1} + 0.05z^{-2}}\,. \tag{E16.6.1}$$

(a) In fixed-point number representation, with $B = 32$ bits, and range $(-1, 1)$,

i. calculate the output noise based on PQN,

ii. verify the results with Monte Carlo.

(b) In floating-point number representation with $p = 24$ bits, and sine-wave input in $[-1, 1]$ with frequency $f_1 = 0.015 f_s$,

i. calculate the output noise based on PQN,

ii. verify the results with Monte Carlo.

Hint: follow the guidelines given in Example 12.2.

16.7 Design a cascade implementation in order to realize the following transfer function:

$$H(z) = \frac{0.6}{(1 - 0.8z^{-1})^2}\,. \tag{E16.7.1}$$

Notice that during factorization, the constant multiplier can be split arbitrarily:

$$\begin{aligned} H(z) &= \frac{a}{(1 - 0.8z^{-1})} \cdot \frac{b}{(1 - 0.8z^{-1})} \\ &= \frac{1}{(1 - 0.8z^{-1})} \cdot \frac{0.6}{(1 - 0.8z^{-1})} \\ &= \frac{\sqrt{0.6}}{(1 - 0.8z^{-1})} \cdot \frac{\sqrt{0.6}}{(1 - 0.8z^{-1})}\,. \end{aligned} \tag{E16.7.2}$$

(**a**) In fixed-point number representation, with $B = 32$ bits, and range $(-1, 1)$,

- **i.** distribute the constant weighting (find a and b) to avoid overload but minimize roundoff, assuming that the input is full scale,
- **ii.** calculate the output noise based on PQN,
- **iii.** verify the results with Monte Carlo.

(**b**) In floating-point number representation with $p = 24$ bits, and a sufficient number of exponent bits to avoid overload, with a sine-wave input in $[-1, 1]$ with frequency $f_1 = 0.0101 f_s$,

- **i.** determine how to distribute the constant weighting (find a and b) to avoid overload but minimize roundoff,
- **ii.** calculate the output noise based on PQN,
- **iii.** verify the results with Monte Carlo.

16.8 An allpass filter is given by

$$H(z) = \frac{0.7 + 0.19z^{-1} - 0.28z^{-2} + z^{-3}}{1 - 0.28z^{-1} + 0.19z^{-2} + 0.7z^{-3}} \,. \tag{E16.8.1}$$

The filter can be implemented in different structures. Two examples are shown in the following figures:

(**1**) direct-form II structure (Fig. E16.8.1, with $b_0 = 0.7$, $b_1 = 0.19$, $b_2 = -0.28$, $b_3 = 0.1$, $a_1 = -0.28$, $a_2 = 0.19$, and $a_3 = 0.7$),

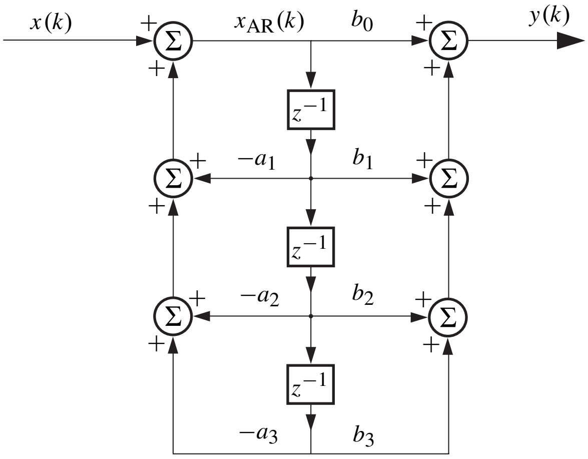

Figure E16.8.1 Direct-form II structure for the filter.

(**2**) lattice realization (Fig. E16.8.2, see also (Jackson, 1996), with $k(1) = -0.4609$, $k(2) = 0.7569$, $k(3) = 0.7000$).

With a program written in Matlab, simulate these filter implementations with different precisions:

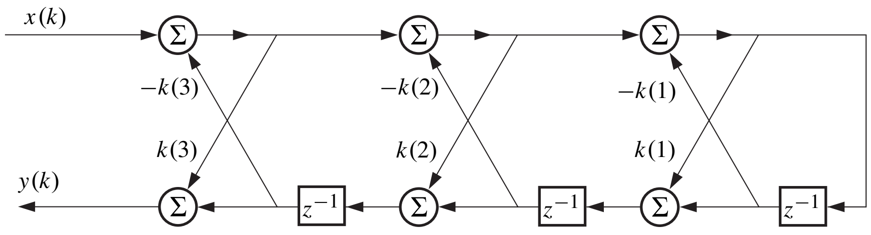

Figure E16.8.2 Lattice structure for the allpass filter.

(a) with double precision, using Matlab's internal number representation,
(b) with fixed-point number representation with $B = 16$ bits in $(-1, 1)$,
(c) with floating-point number representation with $p = 12$ bits.

Assuming that the roundoff error of double-precision representation is negligible, calculate the output roundoff errors of the second and third cases.

i. For a sine wave input with amplitude of $A = 0.06$ and frequency $f_1 = 0.1 f_s$, determine the RMS errors and the SNRs at the output.
ii. Check for each number representation and each structure if there is an overload at any node of the block diagram.

Answer the following questions for fixed-point number representation:

iii. Can the input be increased without causing overload? Determine the SNR achieved by the maximum allowable sine wave.
iv. Which signal frequency allows the lowest output SNR? How large is this SNR?
v. What is the maximum sample value which may occur for x_{AR} or at some other node in the direct-form II realization, and/or somewhere in the lattice, if the input waveform is arbitrary, but $|x(k)| \leq 1$ is assured?
vi. Can you suggest safe input scalings to avoid overflow in each filter realization, for an arbitrary input signal?