# Chapter 2
# DDSM and Applications

## 2.1 Principles of Delta-Sigma Modulation

In order to explain the concept of noise shaping in detail, we start with a stand-alone quantizer (see Fig. 2.1a) with a small number of bits that maps the amplitude of an analog input signal to only a few output levels. We show examples of two types of quantizers in Fig. 2.2, namely mid-tread and mid-rise quantizers [6]. These map an analog signal $v$ to a low resolution output signal $y$ having only five or four discrete levels, respectively. $\Delta$ denotes the quantizer step size in Fig. 2.1a and $k$ is a gain factor, which is 1 in the two examples in Fig. 2.2.

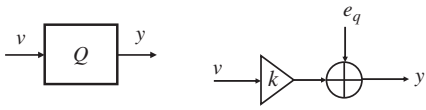
Consider the quantizers shown in Fig. 2.2, which have a step-size $\Delta$. If the input to the mid-tread quantizer is greater than $5\frac{\Delta}{2}$ or less than $-5\frac{\Delta}{2}$, the quantizer saturates at its maximum and minimum output values of $2\Delta$ and $-2\Delta$, respectively. If the input lies between these values (i.e., $|v| \leq 5\frac{\Delta}{2}$), the quantizer is said to operate in the no-overload range. Similarly, the mid-rise quantizer in this example is in its no-overload range when $|v| \leq 2\Delta$.

This coarse input–output mapping (quantization) introduces a quantization error (see Fig. 2.1b) defined by

$$e_q = y - kv, \tag{2.1}$$

where $k$ is a gain factor. In the literature, the quantization error is conventionally called quantization noise [4, 17].

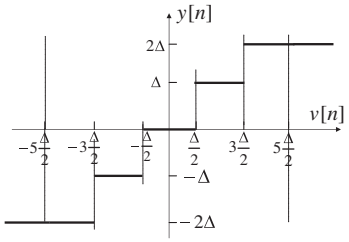
For the quantizers shown in Fig. 2.2, we have illustrated the quantization error $e_q$ as a function of the input $v$ (see Fig. 2.2b, d). In the no-overload range, the quantization noise is bounded in the range $\left[-\frac{\Delta}{2}, \frac{\Delta}{2}\right]$.



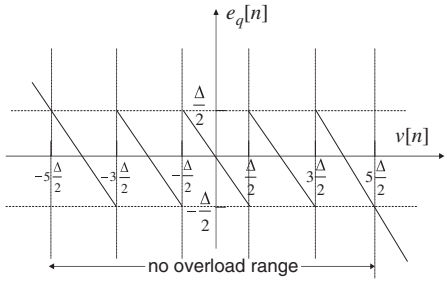

(a) Quantizer (b) Linearized model

**Fig. 2.1** (**a**) A quantizer block diagram and (**b**) its linearized model

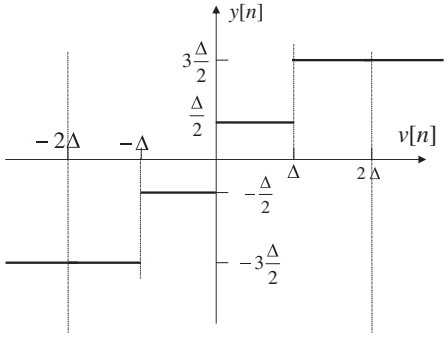
K. Hosseini, M.P. Kennedy, *Minimizing Spurious Tones in Digital Delta-Sigma Modulators*, Analog Circuits and Signal Processing,
DOI 10.1007/978-1-4614-0094-3_2, © Springer Science+Business Media, LLC 2011



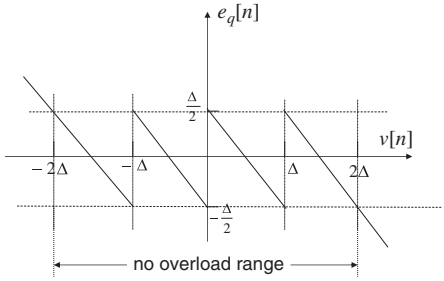

(a) A mid-tread quantizer




(b) Quantization error $e_q$




(c) A mid-rise quantizer




(d) Quantization error $e_q$

**Fig. 2.2** (**a**) Transfer characteristic of a coarse mid-tread quantizer with five output levels. (**b**) The quantization error $e_q$ of the quantizer defined by $e_q = y - kv$ with $k = 1$. (**c**) Transfer characteristic of a coarse mid-rise quantizer with four output levels. (**d**) The quantization error $e_q$ of the quantizer defined by $e_q = y - kv$ with $k = 1$. $\Delta$ is the step-size of the quantizer

The signal-to-quantization-noise ratio (SQNR) is defined as the ratio of the signal power to the power of the quantization noise in a given frequency range, which is typically the signal bandwidth. For a given signal power, coarse quantization (using only a few levels) results in a small SQNR. To increase the SQNR for a given signal power, one must decrease the quantization noise power. This can be achieved by increasing the number of levels in the quantizer. Thus, the larger the required SQNR, the greater the required resolution of the quantizer.

Alternatively, we can use the concept of noise shaping and place the coarse quantizer in a delta-sigma modulator loop, as shown in Fig. 2.3a. Qualitatively, the delta-sigma modulator attenuates the quantization noise power in the signal band and amplifies the out-of-band power. If appropriate filtering is subsequently applied, the out-of-band quantization noise can be attenuated significantly. As we will see, the attenuation of the quantization noise in the signal band results in a larger SQNR compared with the SQNR of a stand-alone coarse quantizer.

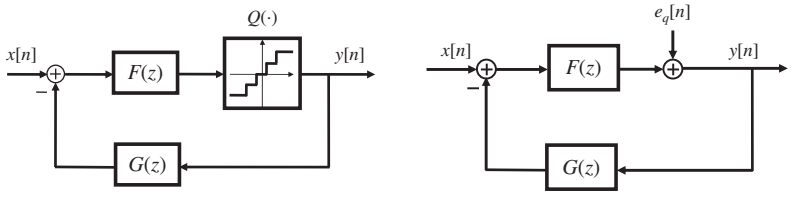
We assume the sampling frequency $f_s$ of the modulator is much greater than twice the bandwidth of the input signal. If a discrete-time signal is sampled at a rate that is much greater than twice its bandwidth, it is said to be oversampled. The oversampling ratio OSR is defined as:

$$\text{OSR} = \frac{f_s}{2f_B}, \tag{2.2}$$

where $f_s$ is the sampling frequency and $f_B$ is the largest frequency component in the signal spectrum. $2f_B$ corresponds to the Nyquist rate [7]; the OSR is the factor by which the sampling frequency exceeds to the Nyquist rate. For example, if $f_B = 44\,\text{kHz}$ and $f_s = 5.632\,\text{MHz}$, then the Nyquist rate is 88 kHz and OSR= 64.

In data converter applications, oversampling and the noise shaping property of the DSM can be used together to achieve a high SQNR using a relatively coarse quantizer; these systems are referred to as oversampled delta-sigma converters [4].

As mentioned earlier, the delta-sigma modulator places the quantizer in a negative feedback loop, as shown in Fig. 2.3a. This architecture contains discrete-time feedback and feedforward filters $G(z)$ and $F(z)$, respectively. In the following, we explain the roles of these filters.




(a) delta-sigma modulator (b) linearized model

**Fig. 2.3** Block diagrams of (**a**) a single-quantizer discrete-time delta-sigma modulator and (**b**) its linearized model with the quantizer gain factor $k = 1$. The input has been oversampled with an oversampling ratio OSR

We can build a simplified linear model of the modulator by replacing the quantizer with a gain factor[1] $k$ and an additive signal source $e_q$, as shown in Fig. 2.3b [4]. Taking the $z$ transforms of the signals in Fig. 2.3b, $Y(z)$ can be written in terms of the $z$-transforms of the input signal $X(z)$ and the quantization noise $E_q(z)$ as follows:

$$Y(z) = \mathrm{STF}(z)X(z) + \mathrm{NTF}(z)E_q(z), \tag{2.3}$$

where

$$\mathrm{STF}(z) = \frac{F(z)}{1 + F(z)G(z)}, \tag{2.4}$$

$$\mathrm{NTF}(z) = \frac{1}{1 + F(z)G(z)}. \tag{2.5}$$

STF($z$) and NTF($z$) are the signal and noise transfer functions, respectively, of the linearized system. These transfer functions are usually designed such that the input signal is not attenuated by the system while the quantization noise is strongly attenuated in the signal band.

In this book, we deal with low pass signals; therefore, the required NTF is high pass in nature, meaning that it attenuates the quantization noise at low frequencies and passes the quantization noise at high frequencies. For example, assume that

$$F(z) = \frac{1}{1 - z^{-1}}, \tag{2.6}$$

$$G(z) = z^{-1}, \tag{2.7}$$

where $F(z)$ is the transfer function of an integrator (or an accumulator in a digital implementation) and $G(z)$ is a simple delay. With these filters, we obtain

$$\mathrm{STF}(z) = 1, \tag{2.8}$$

$$\mathrm{NTF}(z) = 1 - z^{-1}. \tag{2.9}$$

The NTF is a first order high pass filter, as we will see below. In order to understand the behavior of the system in the frequency domain, we write $z = e^{j\omega}$ and calculate the squared magnitudes of the NTF and STF. Note that

$$|\mathrm{STF}(z)|^2_{z=e^{j\omega}} = 1,$$

so the signal passes without any filtering from the input to the output.

[1] In this case, $k = 1$.

For the NTF, we can write

$$\begin{aligned}\mathrm{NTF}(e^{j\omega}) &= 1 - e^{-j\omega} \\ &= 1 - \cos(\omega) + j\sin(\omega),\end{aligned}$$

and

$$\begin{aligned}\left|\mathrm{NTF}(e^{j\omega})\right|^2 &= (1-\cos(\omega))^2 + \sin^2(\omega) \\ &= 2 - 2\cos(\omega) \\ &= 2\left(2\sin^2\left(\frac{\omega}{2}\right)\right) \\ &= \left(2\sin\left(\frac{\omega}{2}\right)\right)^2. \end{aligned} \tag{2.10}$$

The magnitude response of the noise transfer function is thus $2\left(\sin\left(\frac{\omega}{2}\right)\right)$. The peak of this function occurs at $\omega = \pi$ with a magnitude of 2. We are interested in low pass signals so let us examine the function around $\omega \approx 0$. For small $\omega$,

$$2\left(\sin\left(\frac{\omega}{2}\right)\right) \approx 2\left(\frac{\omega}{2}\right) = \omega.$$

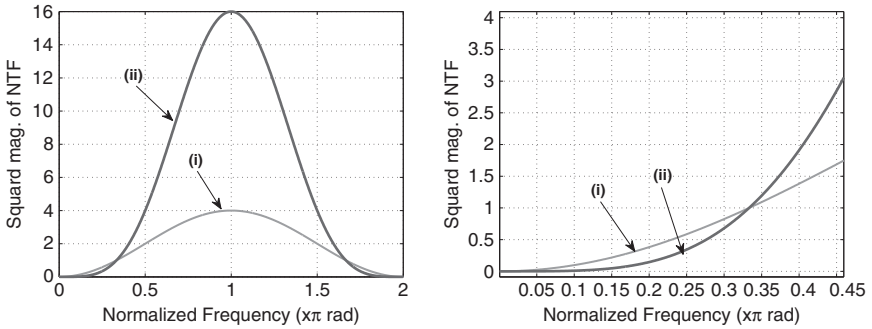
Decreasing $\omega$ decreases the magnitude of the frequency response. At zero frequency, the magnitude of the frequency response becomes zero. We conclude that the delta-sigma modulator in this example implements quantization noise shaping with a high pass characteristic, meaning that it attenuates the noise at low frequencies ($\omega \approx 0$) and amplifies the noise at high frequencies[2] ($\omega \approx \pi$).

Figure 2.4a shows $\left|\mathrm{NTF}(e^{j\omega})\right|^2$ with $\omega$ in the range $[0\ 2\pi]$ for two cases: (i) $\mathrm{NTF}(z) = (1 - z^{-1})$; (ii) $\mathrm{NTF}(z) = \left(1 - z^{-1}\right)^2$. Note that the sampling frequency $f_s$ maps to $2\pi$.

### *2.1.1 SQNR*

Now that we have explained the noise shaping property of the modulator with the help of the linearized model in the first order modulator, we would like to calculate its SQNR by making simplifying assumptions about the statistics of the quantization noise. In practice, these assumptions may not hold in all cases. In particular, as we will see in the next chapters, one of the drawbacks of the white noise approximation is that it does not allow us to predict spurious tones in DSMs. Nevertheless, it allows one to estimate how the SQNR can be improved by increasing the modulator order

[2] Note that $\omega = 2\pi\frac{f}{f_s}$ [7]; therefore, the sampling frequency $f_s$ is mapped to $2\pi$.




(a) First and second order spectrum (b) Zoomed at low frequencies

**Fig. 2.4** (**a**) The squared magnitude of the noise transfer function in a first order [plot (*i*)] and a second order [plot (*ii*)] delta-sigma modulator. (**b**) Zooms of the plots shown in (**a**) at low frequencies. The second order modulator provides greater attenuation of the quantization noise at low frequencies, but amplifies it at high frequencies

and the OSR. When we refer to the white noise approximation throughout this book, we will assume that the following assumptions are valid unless otherwise stated.

We assume that:

- the quantization noise is uniformly distributed in the range $\left[-\frac{\Delta}{2}\ \frac{\Delta}{2}\right]$ and has zero mean;
- it is independent of the input;
- delayed versions of the noise are uncorrelated with each other.

The last condition implies that $R_{ee}[l_g] = \sigma_e^2 \delta[l_g]$ [7], where $R_{ee}[l_g]$ is the autocorrelation function of $e_q$, $\sigma_e^2$ is the variance of $e_q$, and $l_g$ is a given lag.

First we calculate the power of the quantization noise. The variance of $e_q$ (the average power of $e_q$) is calculated as follows [7]

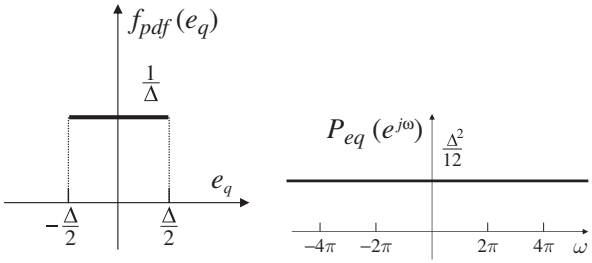
$$\sigma_e^2 = E[e_q^2] = \int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} f_{pdf}(e_q) e_q^2 de_q = \int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} \frac{1}{\Delta} e_q^2 de_q = \frac{\Delta^2}{12},$$

where $f_{pdf}(e_q) = \frac{1}{\Delta}$ is the probability density function of $e_q$ with the uniform distribution shown in Fig. 2.5a.

The autocorrelation function of $e_q$ for a lag $l_g$ is given by

$$R_{ee}[l_g] = \sigma_e^2 \delta[l_g],$$

which has a nonzero value of $\frac{\Delta^2}{12}$ at zero lag; otherwise, it is zero. The power spectral density of $e_q$ is the discrete-time Fourier transform of the autocorrelation function [7]. This results in a white power spectral density $P_{eq}(e^{j\omega}) = \sigma_e^2$, as shown in Fig. 2.5b, with a constant amplitude of $\sigma_e^2 = \frac{\Delta^2}{12}$.




(a) PDF of $e$ (b) PSD of $e$

**Fig. 2.5** (**a**) Probability density function (PDF) of the quantization noise uniformly distributed in $\left[-\frac{\Delta}{2}\ \frac{\Delta}{2}\right]$. (**b**) Power spectral density (PSD) of the quantization noise. The PSD is the discrete-time Fourier transform of the autocorrelation function $R_{ee}[l_g]$ [7]

If the above assumptions hold, the filtered quantization noise has the same shape as the NTF with a scaling factor of $\sigma_e^2$. Hence,

$$P_{e_{sh}}\left(e^{j\omega}\right) = |\text{NTF}|^2 \left|P_{eq}\left(e^{j\omega}\right)\right| = \left(2\sin\left(\frac{\omega}{2}\right)\right)^2 \sigma_e^2,$$

where $P_{e_{sh}}$ is the PSD of the shaped quantization noise. Next, we estimate the SQNR for this first order modulator when the white noise approximation is valid. Assuming that the sampling frequency is $f_s = \text{OSR} \times\ 2f_B$ and that $f_s$ is mapped at $2\pi$, the upper edge of the signal band will be located at $\frac{\pi}{\text{OSR}}$. The total quantization noise power within the signal bandwidth $\left[-\frac{\pi}{\text{OSR}}\ \frac{\pi}{\text{OSR}}\right]$ is calculated as follows:

$$\sigma_{e_{sh}}^2 = 2\frac{1}{2\pi}\int_0^{\frac{\pi}{\text{OSR}}} \sigma_e^2\left(2\sin\left(\frac{\omega}{2}\right)\right)^2 d\omega \tag{2.11}$$

$$= \frac{8}{\pi}\sigma_e^2\left(\frac{\pi}{4\text{OSR}} - \frac{1}{4}\sin\left(\frac{\pi}{\text{OSR}}\right)\right). \tag{2.12}$$

Expanding the second term as a Taylor series, we obtain

$$\sin\left(\frac{\pi}{\text{OSR}}\right) \approx\ \frac{\pi}{\text{OSR}} - \frac{\left(\frac{\pi}{\text{OSR}}\right)^3}{6},\ \text{for}\ \frac{\pi}{\text{OSR}} << 1. \tag{2.13}$$

Substituting (2.13) into (2.12) yields

$$\sigma_{e_{sh}}^2 \approx\ \sigma_e^2 \frac{\pi^2}{3}\frac{1}{(\text{OSR})^3}. \tag{2.14}$$

The SQNR is defined by

$$\text{SQNR} = \frac{\text{signal power}}{\text{quantization-noise power in the signal band}} \tag{2.15}$$

$$= \frac{P_{x,ave}}{\sigma_{e_{sh}}^2} \tag{2.16}$$

$$\approx \frac{P_{x,ave}}{\sigma_e^2}\frac{3}{\pi^2}\text{OSR}^3 \tag{2.17}$$

It is common to measure SQNR in decibels. Thus,

$$\text{SQNR}_{\text{dB}} \approx 10\log_{10}\frac{P_{x,ave}}{\sigma_e^2} - 10\log_{10}\left(\frac{\pi^2}{3}\right) + 30\log_{10}(\text{OSR}). \tag{2.18}$$

If the oversampled signal is applied directly to the stand-alone quantizer without the delta-sigma loop, the SQNR in dB is given by [5]

$$\text{SQNR}_{\text{dB}} = 10\log_{10}\frac{P_{x,ave}}{\sigma_e^2} + 10\log_{10}(\text{OSR}). \tag{2.19}$$

Note that doubling the OSR increases the SQNR by 3 dB in the stand-alone quantizer, while doubling the OSR results in an increase of approximately 9 dB in the SQNR of a first order delta-sigma modulator. This is why noise shaping helps to improve the resolution of a quantizer.

We can use higher order modulators to obtain a larger SQNR. In the case of a second order modulator (see Fig. 2.4), the SQNR is given by [5]

$$\text{SQNR}_{\text{dB}} \approx 10\log_{10}\frac{P_{x,ave}}{\sigma_e^2} - 10\log_{10}\left(\frac{\pi^4}{5}\right) + 50\log_{10}(\text{OSR}). \tag{2.20}$$

Every doubling of the OSR in the second order modulator results in an increase of approximately 15 dB in the SQNR. In general, for an $l$th order modulator with $\text{NTF}(z) = \left(1 - z^{-1}\right)^l$, the SQNR increases by $(6l + 3)$ dB for every doubling of the OSR [6].

$\text{NTF}(z) = \left(1 - z^{-1}\right)^l$ is not the only possible filter choice in the design of delta-sigma modulators. Locating the zeros and poles of the NTF by design allows one to prescribe the inband SQNR, the modulator stability, and the spectral performance of the modulator (see, for example [6, Chap. 4]).

In this section, we reviewed the concept of noise shaping and explained the effect on the NTF, and consequently on the SQNR, of increasing the order of the modulator. Next, we will look briefly at the classification of modulators based on the types of signals they process, and describe some practical applications.

## 2.2 Classification of Delta-Sigma Modulators

Depending on the discretization of the time and amplitude axes of the input signal, modulators can be classified into the following three categories:

- Continuous-time (CT),
- Discrete-time, continuous-amplitude (DT analog),
- Discrete-time discrete-amplitude (DT digital).

### *2.2.1 Continuous-Time (CT) Analog Modulator*

We consider two distinct implementations: sampled and unsampled quantizers.

#### 2.2.1.1 Case 1: Sampled Quantizer: Synchronous Modulator

In this case, the input is a continuous-time analog signal and the modulator is implemented using continuous-time analog filters. The most common application is in continuous-time (CT) delta-sigma analog-to-digital converters (ADC) [6–21]. In a CT delta-sigma ADC, there is no need for an anti-aliasing filter or a front-end sampler. This simplifies system design by eliminating the anti-aliasing filter, which must precede other types of ADCs. In addition, the use of a CT filter postpones the inevitable sampling of the signal, which takes place at the output of the loop filter instead. Thus, imperfections in the sampling process arise at a much less sensitive point in the loop (see Fig. 2.6a) where the errors at this point are shaped by the NTF of the modulator [6].

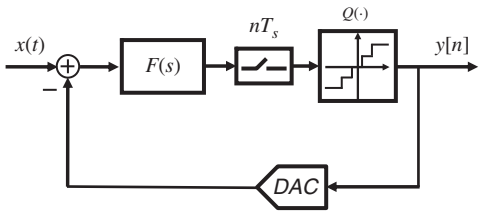
The practical limit on the clock rate of a CT modulator is determined by the regeneration time of the quantizer and the update rate of the feedback DAC, whereas the clock rate in a discrete-time modulator is limited by the op-amp settling requirements [6]. In practice, a CT modulator can operate with a clock frequency which is 2–4 times greater than that which can be achieved with discrete time techniques [6].

#### 2.2.1.2 Case 2: Unsampled Quantizer: Asynchronous Modulator

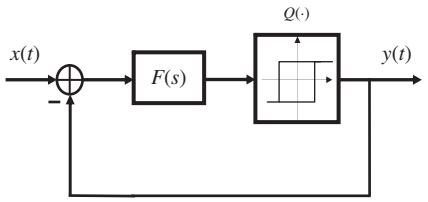
Asynchronous delta-sigma modulators can be used to convert an analog CT input signal into a CT discrete-amplitude output signal (see Fig. 2.6b). In this case, the information in the amplitude of the input signal is coded in the pulse widths of the output signal [22]. Due to its fully analog nature, the asynchronous delta-sigma modulator has a specific application area where pure analog processing is required, such as ADSL/VDSL line drivers, line drivers for optical cables, and UMTS transmitters [22]. In addition, asynchronous ADCs have been reported [23, 24].

### *2.2.2 Discrete-Time (DT) Analog Modulator*

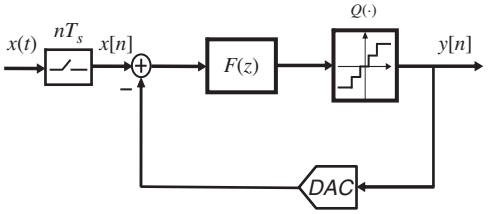
In the case of a discrete-time analog modulator, the input is a sampled analog signal and the modulator uses a discrete-time filter implemented with switched-capacitor




(a) Synchronous CT modulator




(b) Asynchronous CT modulator




(c) DT modulator

**Fig. 2.6** Block diagrams of (**a**) a synchronous continuous-time modulator, (**b**) an asynchronous continuous-time modulator and (**c**) a discrete-time modulator

circuits. The main application of the DT analog delta-sigma modulator is in delta-sigma ADCs [6, 25, 26].

The CT and DT analog categories are beyond the scope of this book and the interested reader may refer to [6] for an excellent exposition of these types of modulators.

### *2.2.3 Discrete-Time Digital Modulators*

In the case of a DT discrete-amplitude modulator, the input to the modulator is a quantized (digital) signal; consequently, the filters are implemented using a finite state machine. In this class, the delta-sigma modulator is implemented with digital circuits and we refer to this type of modulator as a digital delta-sigma modulator (DDSM). The main applications for this class of modulator are delta-sigma DACs [27–33] and delta-sigma fractional-N frequency synthesizers [34–42].

In DACs, the input can be a constant (DC) or a time-varying signal; in general, the input is time-varying. In fractional-N frequency synthesizers, the input to the DDSM

is often a constant digital word. In this case, the DS-based frequency synthesizer is typically used as a local oscillator to generate channel frequencies in a transceiver. The synthesizer can also be used as a stand-alone transmitter that is capable of phase and frequency modulation; in this case, the input to the DDSM is a modulated data stream [43].

In this book, we are interested primarily in the case where the input to the DDSM is a constant digital word; this covers delta-sigma fractional-N synthesizers in the frequency generation application.

Before reviewing the application of DDSMs in DACs and fractional-N synthesizers, we first describe three common DDSM architectures [4, 6].

## 2.3 DDSM Architectures

Three types of modulators are investigated in this book: (1) single-quantizer DDSMs, (2) Error feedback modulators and (3) Multi stAge noise SHaping (MASH). In the following subsections, we will provide an overview of each of these architectures in turn.

### *2.3.1 Single Quantizer DDSMs*

The general block diagram of a single-quantizer DDSM (SQ-DDSM) is shown in Fig. 2.7a. There is only one quantizer in the loop; hence the name "single quantizer" (SQ) DDSM. An example transfer characteristic of the digital multi-level quantizer $Q$ is shown in Fig. 2.7b. The relationship between $y$ and $v$ is described by:

$$y = Q(v) = \begin{cases} R_{lo} + \frac{M}{2}, & v < R_{lo} \\ M\left\lfloor \frac{v}{M} + \frac{1}{2} \right\rfloor, & R_{lo} \le v < R_{hi} \\ R_{hi} - \frac{M}{2}, & v \ge R_{hi}, \end{cases} \tag{2.21}$$

where $\lfloor x \rfloor$ denotes the largest integer less than or equal to $x$, $M$ is a positive even integer referred to as the step-size, and $R_{lo} < R_{hi}$ are arbitrary odd multiples of $\frac{M}{2}$. For the characteristic shown in Fig. 2.7b, $R_{lo} = -\frac{5}{2}M$ and $R_{hi} = \frac{5}{2}M$.

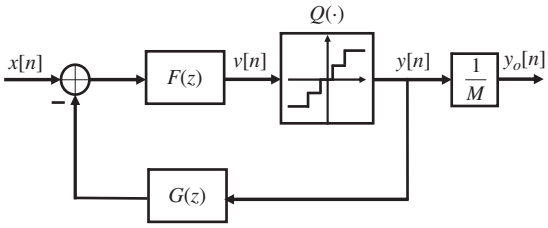
The signal and the noise transfer function of the modulator in Fig. 2.7a are

$$\mathrm{STF}(z) = \frac{F(z)}{1 + F(z)G(z)}, \tag{2.22}$$

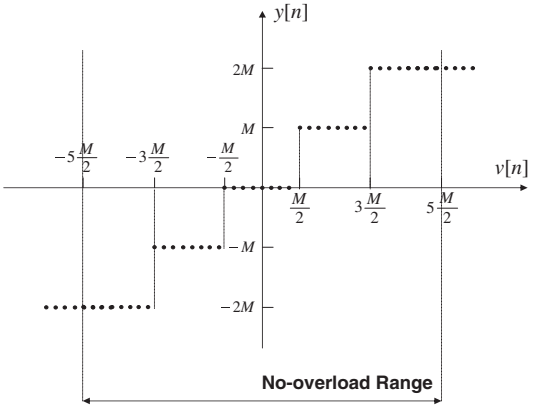
and

$$\mathrm{NTF}(z) = \frac{1}{1 + F(z)G(z)}, \tag{2.23}$$

respectively.




(a) SQ-DDSM




(b) Digital mid-tread quantizer

**Fig. 2.7** (**a**) Block diagram of a multi-bit SQ-DDSM, (**b**) transfer characteristic of an example five level digital mid-tread quantizer with step size $M$

The filters characteristics $F(z)$ and $G(z)$ determine the signal and noise transfer functions and the order of the modulator. A special case is $F(z) = z^{-l}(1 - z^{-1})^{-l}$ and $G(z) = z^l - (z - 1)^l$. These result in

$$\mathrm{STF}(z) = z^{-l}, \tag{2.24}$$

and

$$\mathrm{NTF}(z) = (1 - z^{-1})^l, \tag{2.25}$$

respectively.

Note that the STF is a delay of order $l$ and the NTF is a high pass filter of order $l$. The squared magnitude of the frequency response of the NTF is $\left(2\sin\left(\frac{\omega}{2}\right)\right)^{2l}$. At low frequencies ($\omega \approx 0$) this can be approximated by $\omega^{2l}$. Therefore, the slope of the PSD is 60 dB/decade around $\omega \approx 0$ if $l = 3$.

In modulators of this type with order $l$ greater than 1, the input range over which the quantizer is not overloaded[3] is a fraction of the full scale [4, 6, 44]. A third

[3] The stability of a DSM is often described in terms of the quantizer not being overloaded.

order modulator with the STF and NTF defined by Eqs. (2.24) and (2.25) can be overloaded if the quantizer has only one bit (i.e. two levels) [5]. By providing a larger number of quantizer levels, the input range over which the modulator is not overloaded increases. As an example, if $l = 3$ and $2^{l+1}+1$ output levels are allowed, then an input to the modulator which is less than half of the quantizer's full scale range is sufficiently small to prevent overload of the quantizer [6].

In order to maintain a higher order single-bit modulator in the no-overload region, the useful range of the delta-sigma modulator input signal must be reduced and more poles and zeros must be introduced within the feedback loop, compared to a multi-bit design with a comparable dynamic range [6, 45, pp. 23–33].

The classical SQ-DDSM is characterized by output feedback, i.e. the output $y$ is fed back via the feedback network $G(z)$; an alternative architecture uses error feedback.

### 2.3.2 Error Feedback Modulators

In an Error Feedback Modulator (EFM), the quantization error is calculated and it, rather than the output, is fed back to the input via a filter $H$; that is why it is called *error* feedback. In fact, an EFM is a SQ-DDSM but it is often distinguished as a separate class.

In Fig. 2.8, we show the block diagram of a higher order EFM that uses a multi-bit quantizer (with a transfer characteristic such as that shown in Fig. 2.7b). Similar to the SQ-DDSM, a multi-bit quantizer is needed to maximize the dynamic range in a higher order modulator [6]. The filter $H$ can have a general form, as in an SQ-DDSM. A special case that we consider is when $H(z) = 1 - (1 - z^{-1})^l$. This results in a NTF of the form $(1-z^{-1})^l$ like the SQ-DDSM discussed in Section 2.3.1.

### 2.3.3 MASH Topology

By contrast with the SQ-DDSM and EFM, both of which use a single quantizer, the multistage noise shaping (MASH) technique allows one to realize higher order noise shaping using lower order modulators. In the MASH DDSM, one can use lower order modulators (with orders as low as 1) in a cascade. If first order stages are available, $l$ stages can be combined to form an $l$th order MASH modulator. For

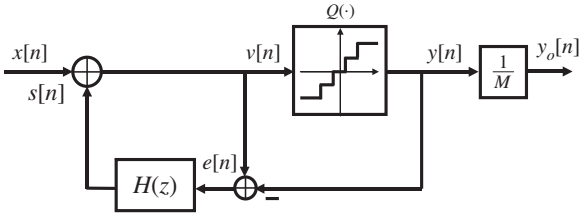



**Fig. 2.8** Block diagram of an error feedback modulator

example, a third order MASH modulator can be implemented using three first order stages in cascade (denoted MASH 1-1-1). Alternatively, two stages can be used if one of the stages uses a second order SQ-DDSM (denoted MASH 1-2 or MASH 2-1). In the latter cases, the stability of the modulator is typically determined by that of the second order modulator.

A MASH DDSM with 1-bit internal first order modulators is a feedforward structure and is unconditionally stable [4]; this is the principal advantage of the MASH modulator over the SQ-DDSM topology. Furthermore, in a MASH DDSM, the stable input range is equal to the full scale while the stable input range is only a fraction of the full scale in an SQ-DDSM. The MASH DDSM is widely used in commercial fractional-N frequency synthesizer products.

Figure 2.9 shows the block diagram of a MASH DDSM where the stages in cascade use first order 1-bit error feedback modulators (denoted EFM1). The EFM1 block is shown in Fig. 2.10a and is equivalent to a first order delta-sigma modulator. The EFM1 modulator uses a 1-bit quantizer with the transfer characteristic shown in Fig. 2.10c. If the threshold point $M$ is a power of 2, the EFM1 can be simply implemented using a conventional digital accumulator, as will be explained in detail in Chapter 3.

To obtain the $l$th order modulator in Fig. 2.9, $l$ $EFM1$ stages are connected in cascade. The input $x$ is applied to the first stage and the inverted quantization error of each stage is fed to the input of the following stage. The 1-bit outputs $y_i$ of the stages are applied to a filter called the noise cancellation network. The function of the noise cancellation network is to eliminate the quantization noise contributions of all of the stages except the last one. In this way, the output contains information related to the input plus shaped quantization noise only from the last stage. The order of the noise shaping filter is equal to $l$. For example, consider the NTF and STF of a third order modulator. The linear model of the EFM1 shown in Fig. 2.10b is used to calculate the STF and NTF of the MASH DDSM. After straightforward calculations, one can show that $E(z) = -ME_q(z)$ where $E_q$ is an additive error source with a non-zero mean $\left(-\frac{1}{2}\frac{M-1}{M}\right)$ in the range $\left\{0, -\frac{1}{M}, \cdots, -\frac{M-1}{M}\right\}$. For the three stages of the MASH 1-1-1 DDSM shown in Fig. 2.9 we write:

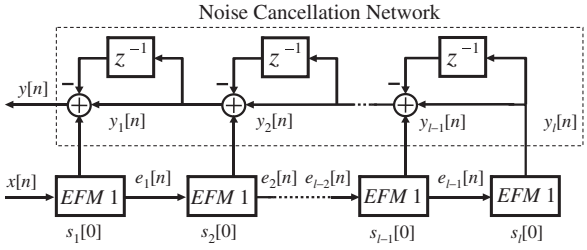
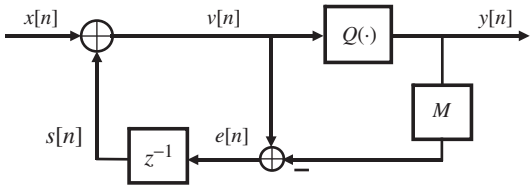
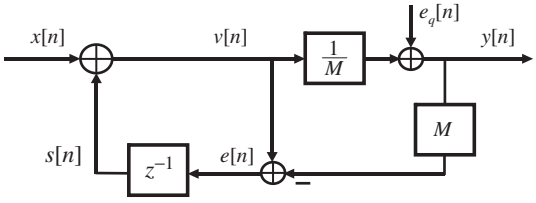


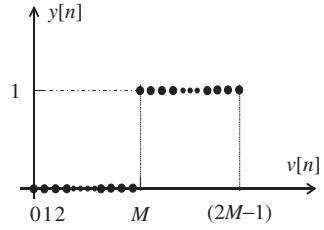

**Fig. 2.9** Block diagram of a MASH DDSM comprising first order error feedback modulators (EFM1) of the type shown in Fig. 2.10a




(a) EFM1




(b) Linearized Model




(c) 1-bit Quantizer

**Fig. 2.10** Block diagrams of (**a**) a first order EFM (**b**) its linearized model and (**c**) transfer characteristic of the 1-bit quantizer used in EFM1

$$Y_1(z) = \frac{1}{M}X + (1 - z^{-1})E_{q1}(z) \tag{2.26}$$

$$Y_2(z) = \frac{1}{M}\big(-ME_{q1}(z)\big) + (1 - z^{-1})E_{q2}(z) \tag{2.27}$$

$$Y_3(z) = \frac{1}{M}\big(-ME_{q2}(z)\big) + (1 - z^{-1})E_{q3}(z), \tag{2.28}$$

where $Y_i(z)$ and $E_{qi}(z)$ are the z-transforms of the outputs $y_i$ and the quantization errors, respectively.

The noise cancellation network multiplies the last equation by $(1 - z^{-1})^2$, the second equation by $(1 - z^{-1})$, the first equation by 1, and adds them together to produce:

$$Y(z) = Y_1(z) + (1 - z^{-1})Y_2(z) + (1 - z^{-1})^2 Y_3(z) \tag{2.29}$$

$$= \frac{1}{M}X(z) + (1 - z^{-1})^3 E_{q3}(z). \tag{2.30}$$

In this way, the quantization noise components $E_{q1}$ and $E_{q2}$ are cancelled exactly and the component $E_{q3}$ is shaped by a third order high-pass filter[4] $(1 - z^{-1})^3$. MASH modulators are popular in applications such as fractional-N synthesis due to the simplicity of their implementation and their inherent stability [44].

Next, we consider the use of the DDSM in two application domains: digital-to-analog conversion and frequency synthesis.

## 2.4 Delta-Sigma DAC

In the following subsection, we study briefly the principles of operation of DS DACs [27–33] which are commercially as important as their ADC counterparts, and their implementation is often as difficult as the implementation of a delta-sigma ADC [6]. A high resolution (such as 18-bit) DAC with a relatively low voltage power supply is implemented effectively using the oversampling and noise shaping concepts. These techniques ease the design of the analog parts such that 18-bit accurate analog components are not required. Such resolution cannot be achieved in a conventional DAC without expensive analog trimming and/or an extremely long conversion time. The price one pays is the introduction of additional digital hardware operating at high frequencies (much higher than the signal bandwidth).

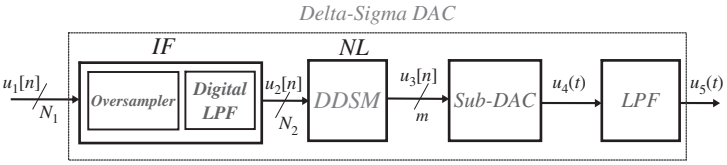
Figure 2.11a shows a typical block diagram of a delta-sigma DAC. Referring to Fig. 2.11, we will explain its operation in the frequency domain. The idea is not to convert the input bits to an analog signal directly, but to truncate the input digital word to a smaller number of bits. The truncated digital signal can then be converted to an analog signal with significantly less complex analog circuitry. The truncation is performed by the DDSM.

For simplicity, we have shown all the digital and analog spectra using the same $x$-axis, $f$. A bandlimited digital low pass signal $u_1$ with bandwidth $2f_B$, $N_1$-bit resolution and sampling frequency $f_{s1} = kf_B$ $(k \geq 2)$ is applied to the system.

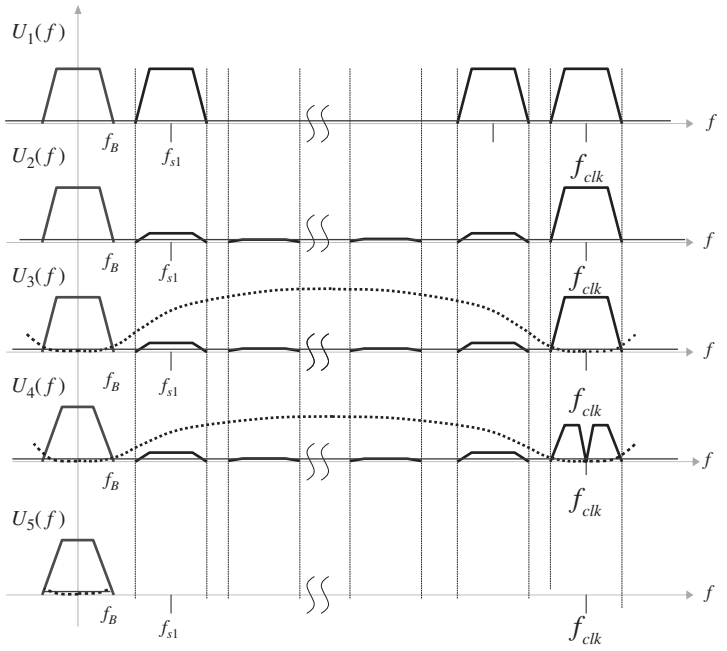
The spectrum of the high resolution digital signal $u_1$ contains the original baseband portion and its replicas located at integer multiples of $f_{s1}$, plus a small amount of quantization noise shown as a solid line in Fig. 2.11b. The interpolation filter (denoted IF), re-samples the digital input signal at a rate $f_{clk}$ which is much greater than the original sampling frequency $f_{s1}$ and applies the resulting oversampled signal to a digital low pass filter. The digital filter attenuates the images located between the baseband and $f_{clk}$.

The oversampled output signal $u_2$, with the resolution $N_2$-bits, is applied to the noise-shaping loop (NL), which is implemented as a DDSM. The images are attenuated by the digital filter in the IF block, relaxing the job of the CT reconstruction filter located after the sub-DAC. The DDSM truncates its $N_2$-bit input $u_2$ coarsely, resulting in $u_3$, which has a smaller number of bits (shown as $m$ in the figure). The

[4] Note that the DC term due to non-zero-mean quantization noise is removed by the high pass filtering applied to the quantization noise.




(a) Delta-sigma DAC block diagram




(b) Signal spectra in a DS DAC

**Fig. 2.11** (**a**) Block diagrams of a delta-sigma DAC and (**b**) associated spectra of the delta-sigma DAC

quantization noise of the modulator is shaped so that most of its power is attenuated around the signal band, as illustrated by the dotted curve in the figure.

The fact that $u_3$ has only a few bits (at the minimum, only one bit) simplifies significantly the subsequent analog stages. $u_3$ is applied to an $m$-bit sub-DAC and the resulting analog output[5] $u_4$ is applied to an analog low pass reconstruction filter (LPF) to remove the quantization noise introduced by the DDSM, giving the desired analog output $u_5$.

After this short introduction, we report a few example specifications in Table 2.1. All have used multi-bit quantizers instead of a 1-bit quantizer for better stability

[5] We have assumed that the DAC transfer function with a zeroth order sample and hold is a sinc function defined by $\frac{\sin(\pi/(\mathrm{OSR}_{f_{s_1}}))}{\pi/(\mathrm{OSR}_{f_{s_1}})}$.

**Table 2.1** Comparison of some delta-sigma DACs

| Author | Hamasaki [27] | Adams [28] | Fujimori [30] | Annovazi [31] | Colonna [32] | Nguyen [33] | Lee [46] |
|---|---|---|---|---|---|---|---|
| Year | 1996 | 1998 | 2000 | 2002 | 2005 | 2008 | 2009 |
| Technology (CMOS μm) | 0.6 | 0.6 | 0.5 | 0.35 | 0.13 | 0.18 | 0.35 |
| DDSM order | 3 | 2 | 3 | 3 | 3 | 2 | 3 |
| DDSM type | NA | NA | SQ | SQ | SQ | NA | SQ |
| NTF type | NA | NA | Specific | Chebyshev | Specific | NA | Specific |
| Quantizer levels | 5 | 64 | 31 | 13 | 17 | 256 | 7 |
| SNDR (dB) | 90 | 100 | 102 | 86 | 88 | NA | 69 |
| DR (dB) | 100 | 113 | 120 | 98 | 97 | 108 | 88 |
| VDD (V) | 3 | 5 | 5 | 3.3 | 3.3 | 1.8 | 0.8 |
| Analog Power (mW) | 10 | NA | 125 | 16.3 | 6.825 | 0.7 | NA |
| Digital Power (mW) | 12 | NA | 10 | 11.55 | 0.375 | 0.4 | NA |
| Total power (mW) | 22 | 125 | 155 | 27.85 | 7.25 | 1.1 | 1.3 |
| Area ($mm^2$) | 3.07 | 9.92 | 7.8 | 1.45 | 0.22 | NA | 1.76 |

NA indicates "Not Available".

and lower quantization noise.[6] Third order modulators are used in all cases except in [28, 33] where second order modulators are used. In these two cases, 64 and 256 quantizer levels were used to compensate for the reduction of SQNR with the lower modulator order. In most cases, the modulator is constructed using a single quantizer and the NTF is different from one design to another by having zeros at different frequencies. The typical dynamic range for audio applications is about 100 dB. One work [30] achieved 120 dB, but with significantly higher power consumption (155 mW [30]).

## 2.5 Phase-Locked Loop Frequency Synthesizers

The limited bandwidth available to each user in wireless systems mandates the precise definition of the carrier frequencies in both the transmit and receive paths. Frequency synthesizers generate periodic signals with accurately defined frequencies, thus serving as an integral part of radio frequency transceivers.

Frequency synthesis continues to be a challenge, fundamentally because performing algebraic operations on frequencies is more difficult than on other electrical quantities such as voltage or current. The challenge has taken different directions through the years, motivating the invention of various architectures and circuit techniques. As RF systems incorporate higher levels of integration, frequency synthesizers must deal with additional trade-offs resulting from application requirements such as monolithic implementation, low cost, minimal number of external components, and low power dissipation [47].

[6] The greater the number of quantizer levels, the smaller quantization error.

In the next subsection, we will first describe the basic idea of frequency synthesis using integer-N phase-locked loops. Then we will explain the principle of operation of a fractional-N synthesizer in order to obtain fine frequency resolution. We will illustrate how the performance of a fractional-N synthesizer can be improved with the aid of a DDSM.

### 2.5.1 Integer-N Frequency Synthesizers

Three architectures [44, 48] are commonly used to synthesize a desired frequency from a reference frequency: (i) table-look-up synthesizers, (ii) direct synthesizers and (iii) phase-locked loop (indirect) synthesizers. The first method cannot be used for high output frequencies, and the second approach is too bulky for silicon integration. Therefore, the phase-locked loop (PLL) is the dominant solution in frequency synthesizers [47] for wireless applications. In this case, the reference frequency is multiplied by a user-defined number. This is achieved by dividing the output frequency by that number, and adjusting the output frequency such that the divided frequency is equal to the reference frequency.

A PLL is a feedback system that operates on the excess phase of nominally periodic signals, i.e., the feedback operation in the loop automatically adjusts the phase of the locally generated signal $y(t)$ to match the phase of the fixed reference signal $x(t)$. As shown in Fig. 2.12, a PLL comprises a phase detector (PD), a low pass filter (LPF), and a voltage-controlled oscillator (VCO). The phase error between $x(t)$ and $y(t)$ is amplified and fed back so as to minimize the phase difference between $x(t)$ and $y(t)$. The loop is considered "locked" if the phase difference is constant; this corresponds to the input reference and output VCO frequencies being equal [49].

In the locked condition, the PLL operates as follows. The phase detector calculates the phase difference between the input reference and the VCO's output signal, and produces an output which is a function[7] of the phase difference. The low pass filter suppresses high frequency components from the PD output. The output of the filter is applied to the VCO to produce the desired output frequency. The VCO oscillates at a frequency that is equal to the input reference frequency but with a constant phase difference. In this way, the filter generates an appropriate control voltage for the VCO.

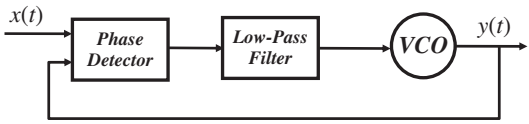



**Fig. 2.12** A general phase-locked loop. VCO stands for voltage-controlled oscillator

[7] Ideally, the PD output is proportional to the phase difference.

Two PLL-based frequency synthesizer architectures are commonly used in applications today, namely integer-N and fractional-N synthesizers. The two implementations differ in how the divider is implemented and controlled. In this section, we discuss integer-N synthesizers. More details of fractional-N synthesizers will be given in later sections.

In Fig. 2.13, the PLL performs frequency multiplication, by means of a negative feedback path, to generate an output frequency $f_{out}$ that is an integer multiple of the reference frequency $f_{ref}$. When the loop is locked,

$$f_{out} = f_{ref} \cdot N. \tag{2.31}$$

A reference frequency is provided to the phase detector for comparison with the divided VCO frequency $f_{div}$. In the "locked state", the VCO frequency is defined by Eq. (2.31). Programming the divider $N$ with a new division number $N$ can change the VCO output frequency, resulting in a frequency $f_{out}$ that can be tuned across the overall band of interest. The primary constraint in this integer-N architecture is that the minimum channel spacing equals $f_{ref}$. As long as the loop is locked, the VCO output will have the same frequency resolution as the reference frequency, which is typically dependent on an external crystal oscillator. For example, with a reference frequency of 30 kHz and a division number of $N = 33000$, the VCO output frequency is 990 MHz. Assuming that the frequency accuracy of the oscillator is 1 ppm, the output of the VCO is accurate to $\pm 990$ Hz around a 990 MHz carrier frequency, and the frequency resolution is 30 kHz.

The architectural simplicity of integer-N PLL frequency synthesizers has made them a popular choice for a variety of telecommunication systems [47]. However, the integer-N architecture has some significant drawbacks.

The frequency resolution, i.e. the channel spacing, is equal to the reference frequency, meaning that only integer multiples of the reference frequency can be synthesized. Therefore, if fine tuning is required, the designer's only choice in an integer-N PLL is to decrease the reference frequency.

Stability requirements limit the loop bandwidth to about one tenth of the reference frequency [45, 50]; therefore, decreasing the reference frequency increases the settling time as the loop bandwidth also has to be decreased. A large settling time is

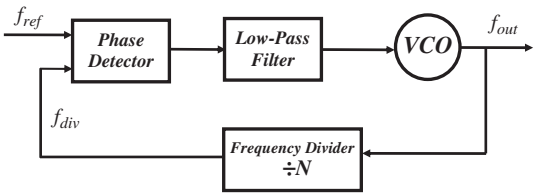



**Fig. 2.13** Block diagram of a PLL-based integer-N frequency synthesizer. It comprises a phase detector, a low-pass filter and a voltage-controlled oscillator in the forward path and an integer frequency divider in the feedback path

not allowed by most communication standards [45]. Also, a reduced loop bandwidth allows less suppression of the VCO's inherent phase noise.

Another drawback of the integer-N PLL is the trade-off between phase noise and settling time when the divider ratio becomes large. The contributions to the output phase noise of almost all PLL building blocks, except the VCO, are multiplied by the division ratio [51]. A high output frequency resolution needs a small input reference frequency, which in turn requires a large divider value. A large divider value increases the inband noise, thereby increasing the rms phase error. In order to decrease the inband noise, the loop-bandwidth has to be kept low; this in turn increases the settling time.

In addition, if a small reference frequency is chosen, the reference spur[8] in the output phase noise is located at a smaller offset frequency. In order to suppress this spur, the loop bandwidth has to be decreased well below $f_{ref}$; this again increases the settling time.

In short, the design of integer-N PLL frequency synthesizers poses a trade-off between frequency resolution, spectral purity, and the PLL's dynamic behavior. An alternative way to obtain high resolution without compromising the dynamic performance is to implement fractional division. This is the topic of the next section, where we explain the principle of operation of a fractional-N frequency synthesizer.

### 2.5.2 Fractional-N Frequency Synthesizers

In fractional-$N$ frequency synthesizers, fractional multiples of the reference frequency can be synthesized, allowing a higher reference frequency for a given frequency resolution. This in turn means that the loop bandwidth can be increased without compromising the spectral purity. Therefore, the PLL dynamics are accelerated and the total amount of capacitance required in the loop filter can be decreased so that single chip integration of the frequency synthesizer becomes feasible.

The basic idea behind fractional-N synthesis is division by fractional ratios, instead of only integer ratios [52, 53]. To accomplish fractional division, the same frequency divider is employed as in an integer-N frequency synthesizer, but the division is controlled differently. In Fig. 2.14, the division modulus of the frequency divider is controlled by the carry output of a simple digital accumulator[9] that is $n_0$-bits wide. To realize a fractional division ratio $N = N_0 + \beta$, with $\beta \in \left\{0, \frac{1}{2^{n_0}}, \frac{2}{2^{n_0}}, \cdots, \frac{2^{n_0}-1}{2^{n_0}}\right\}$, a digital input $X = \beta \cdot 2^{n_0}$ is applied to the accumulator. With the carry output $c$ as the control signal, $X$ ones and $2^{n_0} - X$ zeros are generated for every $2^{n_0}$ output samples. When the carry out is one, the divider value is set to $N_0 + 1$, and the divider value is set to $N_0$ when the carry out is zero.

---

[8] The reference spur refers to unwanted frequency modulation of the VCO at the reference frequency, $f_{ref}$.

[9] As we will see in Chapter 3, the accumulator can be considered as a first order DDSM.

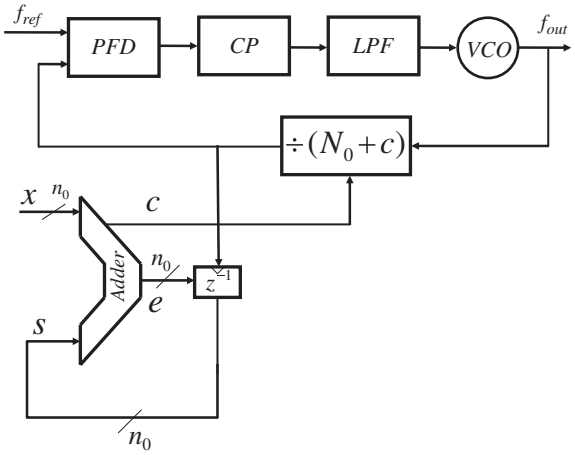



**Fig. 2.14** Block diagram of a fractional-N PLL with a digital accumulator controlling the division ratio. The synthesizer includes a phase frequency detector (PDF), LPF, a VCO, and a variable modulus divider. The carry out of the adder $c$ controls the divider modulus

This means that the frequency divider divides $2^{n_0} - X$ times by $N_0$ and $X$ times by $N_0 + 1$, resulting in a division ratio $N_{mean}$, given by:

$$N_{mean} = \frac{(2^{n_0} - X) \cdot N_0 + X \cdot (N_0 + 1)}{2^{n_0}} = N_0 + \frac{X}{2^{n_0}} = N_0 + \beta. \tag{2.32}$$

Therefore,

$$f_{out} = (N_0 + \beta) f_{ref}. \tag{2.33}$$

In this case, the frequency resolution $\Delta f$ is defined by

$$\Delta f = \frac{1}{2^{n_0}} f_{ref}. \tag{2.34}$$

Equation (2.34) shows that, for a given reference frequency, it is possible to make the frequency resolution arbitrarily high by making the accumulator wordlength sufficiently large. For example, in a DCS-1800 telecommunication system, the channel spacing of 200 kHz can be accommodated by selecting $f_{ref} = 26$ MHz and using an accumulator width $n_0$ of more than 8 bits.

While dynamically switching the divider modulus solves the problem of achieving non-integer multiples of the reference frequency, a price is paid in the form of increased phase noise resulting from jitter in the feedback signal. During each reference period, the difference between the actual modulus ($N_0$ or $N_0 + 1$) and

the desired average modulus $(N_0 + \beta)$ represents a phase error. This error gets injected into the PLL and results in increased phase noise. The amount by which the phase noise is increased depends upon the characteristics of the output of the divider moduli or, equivalently, the spectrum of the signal that controls the divider in the feedback loop.

### 2.5.3 Spurious Tones

In the digital accumulator implementation of a fractional-N synthesizer shown in Fig. 2.14, once the wordlength $n_0$ and input $X$ are fixed, the carry out control signal exhibits periodic behavior. In fact, since the accumulator is a finite state machine, it cycles through its states in a periodic manner, the length of the cycle depending on $X$ and $n_0$. As an example, let us assume that the reference frequency is 1 MHz, $X = 5$, $n_0 = 3$ and $N = 100$. With these values, the output frequency is calculated from Eq. (2.33) as

$$f_{out} = \left(100 + \frac{5}{2^3}\right)1 = 100.625 \text{ MHz}.$$

The signal values of $s$ in the accumulator for 11 cycles of the reference clock are summarized in Fig. 2.15. We assume that the initial state $s[0]$ of the register is zero. The carry out signal $c$ is determined by:

$$c = \begin{cases} 0, & X + s < 8 \\ 1, & X + s \geq 8, \end{cases} \tag{2.35}$$

and the error $e$ is equal to $(X + s)$ modulo 8.

| Clock count | X | s | e | X+s | c |
|---|---|---|---|---|---|
| 0 | 5 | 0 | 5 | 5 | 0 |
| 1 | 5 | 5 | 2 | 10 | 1 |
| 2 | 5 | 2 | 7 | 7 | 0 |
| 3 | 5 | 7 | 4 | 12 | 1 |
| 4 | 5 | 4 | 1 | 9 | 1 |
| 5 | 5 | 1 | 6 | 6 | 0 |
| 6 | 5 | 6 | 3 | 11 | 1 |
| 7 | 5 | 3 | 0 | 8 | 1 |
| 8 | 5 | 0 | 5 | 5 | 0 |
| 9 | 5 | 5 | 2 | 10 | 1 |
| 10 | 5 | 2 | 7 | 7 | 0 |

**Fig. 2.15** The first 11 samples of the signal values in a three bit accumulator with input $X = 5$ and $s[0] = 0$

Note that the internal state $s$ and the carry out signal $c$ are periodic with a period of 8 clock cycles. When $X = 5$, five ones are generated during each period so that the average value of the carry out signal over a period is equal to $\frac{5}{8}$, corresponding to the desired fraction. This enables one to generate the output frequency of 100.625 MHz and steps of $\frac{1}{8} f_{ref} = 0.125\,\text{MHz}$. Each time the carry out signal is unity, it sets the divider value to 101. The fact that the actual divider value (100 or 101) is different from the desired divider modulus (100.625) shows up as an instantaneous phase difference at the input to the phase frequency detector. This phase error is determined by the nature of the carry out signal and causes a phase error (which is periodic in this case). The periodicity in the phase error results in a tonal spectrum. Any tones that are located outside the PLL bandwidth are attenuated by the LPF. However, those that are inside the PLL's bandwidth pass through the LPF and modulate the VCO frequency, manifesting themselves as undesirable spurious tones (so-called "spurs") in the output phase noise.

These tones are also called fractional tones because they are located at fractional multiples of the reference frequency. In our example, the period is 8 reference cycles; therefore, the first tone is located at $\frac{1}{8} f_{ref} = 0.125\,\text{MHz}$, which can be inside the loop bandwidth. Such spurs are not tolerated by most wireless communication standards [44, 47].

One way to attenuate these tones is to decrease the loop bandwidth. However, this solution negates the principal advantage of the fractional-N synthesizer, namely that of having fine tuning resolution while maintaining a relatively large loop bandwidth.

Another way to eliminate fractional tones is to introduce randomness to break the periodicity in the sequence of the division moduli while still achieving the desired average modulus [45]. One can generate a control sequence that approximates a sequence of independent random variables that take on the values 0 and 1 with probabilities of $1 - \frac{X}{2^{n_0}}$ and $\frac{X}{2^{n_0}}$, respectively. During the $n$th reference period, the divider modulus is still $N_0$ or $N_0 + 1$ with the prescribed probabilities. However, the resulting sequence of moduli has the desired average but the power spectral density of the error is spread uniformly over many frequencies. In this way, fractional tones can be eliminated. Instead of tones, this modified technique ideally introduces white noise with a low PSD. Unfortunately, the portion of the white noise within the PLL's bandwidth is integrated by the PLL. Consequently, the overall contribution to the phase noise can be significant unless the PLL bandwidth is small.

Alternatively, one can generate a randomized control sequence whose power at frequencies below the loop bandwidth is highly attenuated [54–56]. A DDSM can generate such a noise shaped control sequence whose power is located mostly outside the PLL loop bandwidth.

An example of a delta-sigma fractional-N PLL is shown in Fig. 2.16. The system comprises a phase frequency detector, a charge pump, a LPF and a VCO in the forward path and a controlled multi-modulus divider in the feedback path. Like the simple accumulator implementation, the DDSM output controls the divider moduli in order to implement fractional division. The input to the DDSM is a constant digital value $X$ and it is clocked by the output of the divider. The DC component of the DDSM's output power spectrum is proportional to $X$; therefore, the time average

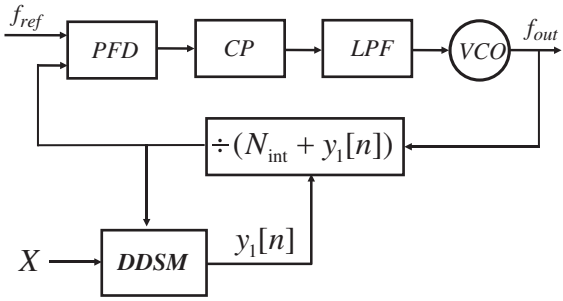



**Fig. 2.16** Block diagram of a delta-sigma fractional-N PLL

of the output sequence $y_1$ is proportional to $X$; this sets the desired fraction. The divider divides the VCO output frequency $f_{out}$ by the integer $N_{int} + y_1[n]$, where $N_{int}$ is a fixed integer and $y_1[n]$ is the DDSM output at each instant $n$. Over many reference cycles, the average of $N_{int} + y_1[n]$ approaches $N_{int} + \beta$, implementing (in a time-average sense) the fractional division $\beta$.

In short, the DDSM with input $X$ is used to perform the following operations:

- Its output signal $y_1$ gives the desired fraction $\beta$ on average.
- $y_1$ possesses a colored spectrum. Firstly, it has a DC tone whose amplitude is proportional to $X$, setting the desired fraction.[10] Secondly, the low frequency part of the rest of its spectrum is attenuated and its high frequency part is amplified. The former is desirable because it rejects the portion of the power spectrum that passes through the LPF. The latter is undesirable; however, it is rejected in principle by the LPF.
- A higher order DDSM[11] randomizes the error sequence with or without one of the auxiliary randomization techniques described in Chapters 3, 4 and 5, to break patterns resulting from short periodic cycles.

In the following, we show by simulation how the delta-sigma modulator can be used to remove fractional tones. Firstly, in Fig. 2.17, we show for comparison the PSDs of the output signals of a digital accumulator and a third order MASH delta-sigma modulator of the type shown in Fig. 2.9. For both configurations, the wordlength is $n_0 = 18$ and the input is $X = 256$. The initial value of the register in the accumulator was set to unity. The initial values of the MASH DDSM were set to 1, 0 and 0, for the first, second and third stages, respectively. As we will see in Chapter 3, the period of the output of the accumulator for this combination of input

[10] Note that the DC term due to non-zero-mean quantization noise is removed by the high pass filtering applied to the quantization noise.

[11] The simple accumulator is a first order DDSM; it fails to perform proper noise shaping and randomization, as we will see in Chapter 3. A higher order modulator is required for adequate randomization of the quantization noise.

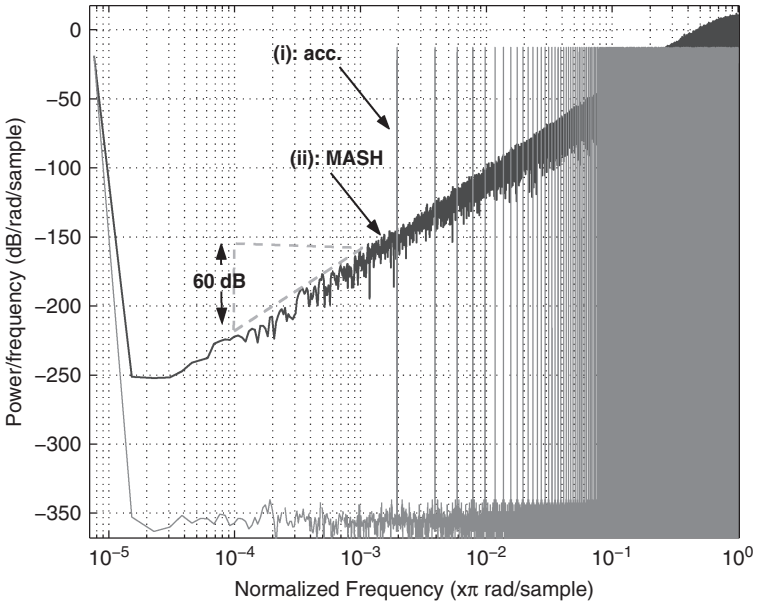


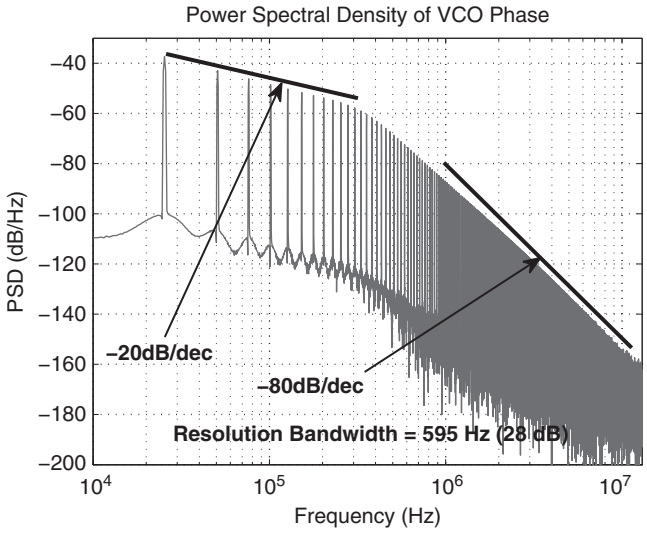

**Fig. 2.17** Spectral plots of the output signals of simple accumulator [plot (*i*)] and, a third order MASH DDSM [plot (*ii*)], both with $n_0 = 18$, $X = 256$. The envelope of the third order DDSM has a slope of 60 dB/decade at low frequencies

and initial conditions is $\frac{2^{18}}{2^8} = 1024$. Therefore, we would expect that the first (fundamental) tone in the accumulator output spectrum should be located at $\frac{2\pi}{1024}$. Plot (i) in Fig. 2.17 confirms that the first tone is located at approximately $2 \times 10^{-3}\pi$, as expected. By contrast, the third order MASH DDSM randomizes the output control sequence and the power of its quantization noise is distributed over many tones so that the output spectrum is smoothly shaped toward higher frequencies, as shown in Fig. 2.17.

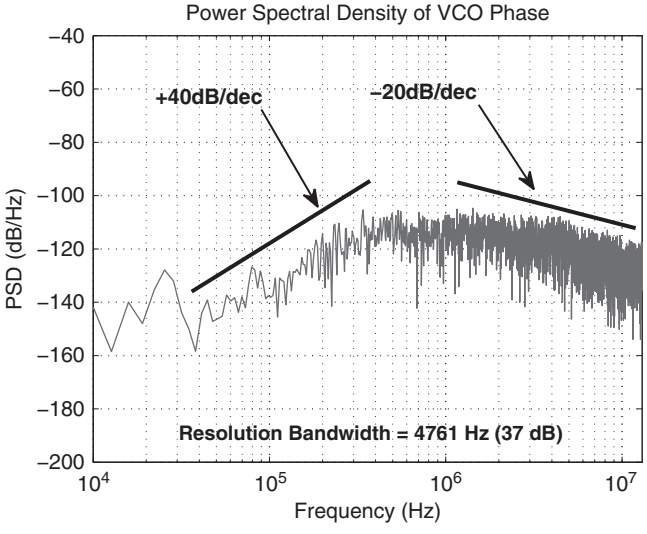
Note that we have plotted the spectra using logarithmic axes, whereas we used linear axes in the introduction to the delta-sigma modulation at the beginning of this chapter. Logarithmic axes are mainly used in the literature for characterizing noise shaping in a delta-sigma modulator. In this example, we have a third order modulator; consequently, the PSD is proportional to $\omega^6$ at lower frequencies. Hence, the slope is 60 dB/decade, as can be seen from the figure.

In this example, the accumulator and the MASH DDSM were implemented in a Verilog-AMS behavioral model of the PLL [57]. The modulators were modelled using Verilog. The PFD, CP, VCO and the divider were described by behavioral Verilog-AMS models. The third order loop filter was implemented based on passive capacitors and resistors. The VCO and the divider blocks were merged into one block in order to decrease the simulation time [51]. The loop parameters are as follow: the reference frequency is 26 MHz; the loop bandwidth is 300 kHz; the LPF

order is three; the input to the modulators is 256; the wordlength of the accumulator and the MASH DDSM is 18; the integer divider is 68; the VCO gain is 70 MHz/V and the charge pump current is 1 mA. For these simulations, we set the up and down currents of the CP to be equal and minimize the noise sources and jitter of the VCO, PFD, CP, reference input and digital blocks to enable us to examine the quantization noise of the modulators in isolation.




(a) Accumulator based




(b) MASH based

**Fig. 2.18** Phase noise plots of (**a**) accumulator and (**b**) MASH based fractional-N PLLs

In Fig. 2.18a, b, we show the resulting phase noise plots. The period of the accumulator output is 1024 reference cycles; therefore, the first fractional tone in the phase noise plot is located at an offset of $\frac{f_{ref}}{1024} \approx 25.4$ kHz. This fractional tone, and tones close to it, lie within the loop bandwidth of the PLL; therefore, they show up in the phase noise plot in Fig. 2.18a with higher power than those located outside the PLL bandwidth. The third order MASH DDSM fixes the problem of fractional tones, as can be seen in Fig. 2.18b. None of the high power fractional tones in the spectrum of the accumulator-based PLL is present in the case of the MASH-based PLL. However, since most of the quantization noise produced by the DDSM is pushed toward higher frequencies, the out-of-band phase noise content is larger than for the accumulator based PLL, falling off at $-20$ dB/decade instead of $-80$ dB/decade.

In this example, the MASH DDSM has been configured in such a way that it exhibits noise shaping without producing fractional tones. In general, however, most DDSM configurations may have reduced performance due to the nature of their digital implementation. In fact, if the modulators are not designed properly or special measures are not taken, they can exhibit qualitatively similar performance to that of the simple accumulator, namely having fractional tones, otherwise called "spurious tones" or spurs.

The accumulated quantization noise appears as a phase error at the PFD input. This accumulation process contributes to a $-20$ dB/decade slope in the phase noise plots. As shown in plot (i) of Fig. 2.17, the slope of the PSD of the accumulator is 0. Due to the prescribed phase conversion, the slope in Fig. 2.18a below the loop bandwidth is $-20$ dB/decade and above the loop bandwidth is $-20 - 60 = -80$ dB/decade. The $-60$ dB/decade term is due to the roll-off of the third order loop filter. By contrast, in the case of the MASH-based PLL, the slope below the loop bandwidth is $+60 - 20$ dB $= +40$ dB/decade and the slope is $+60 - 20 - 60 = -20$ dB/decade above the loop bandwidth.

## 2.6 Simulink Models and MATLAB Codes for DDSMs

In this section, we provide Simulink models and MATLAB code[12] for three sample DDSM architectures, namely SQ, EFM and MASH. First we study the SQ-DDSM.

### *2.6.1 SQ-DDSM*

#### 2.6.1.1 Simulink

Figure 2.19 shows the block diagram of a third order SQ-DDSM [9]. This modulator implements the STF and NTF given in Eqs. (2.24) and (2.25) with $l = 3$, namely $STF(z) = z^{-3}$ and $NTF(z) = (1 - z^{-1})^3$. These can be found by replacing

[12] All MATLAB and Simulink files described in this book are available for download from: http://cas.tyndall.ie

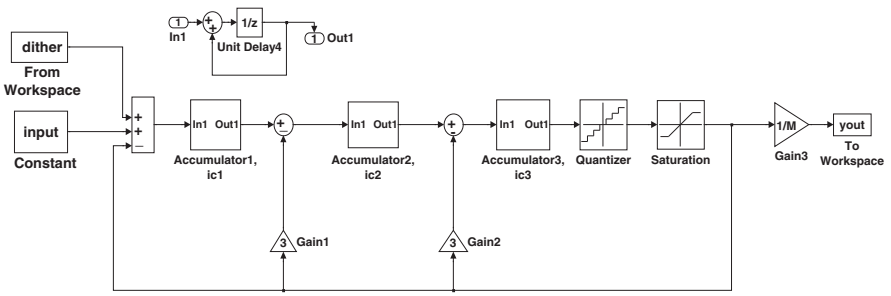



**Fig. 2.19** Block diagram of a third order SQ-DDSM (SQDDSM3rd.mdl) simulated using Simulink

the quantizer and saturation block with a gain factor ($k = 1$) and an additive noise source $e_q$ and finding the output of the saturation block in the $z$ domain as a function of the main input and $e_q$.

The main input to the modulator is a constant value. A pseudorandom binary "dither" sequence generated in MATLAB, taken from the workspace, is added to the constant input to break up periodic cycles, as we will discuss in detail in the next chapter. There are three identical accumulators in the loop. The details of the first accumulator are shown in the figure. The initial value of the internal unit delay is set to $\mathsf{ic}_i$ in $\mathsf{Accumulator}i$. The Quantizer and Saturation blocks implement a mid-tread digital quantizer like the one shown in Fig. 2.7b.

In Table 2.2, we show how each block is configured and in which library it can be found.

The Start time and the Stop time in the Simulation Parameters menu are set to 1 and sim_time. The type in the Solver option is set to "Fixed-step" and "discrete (no continuous step)". This setup assumes that the time index is unit-less integer $(1, 2, 3, \cdots,$ sim_time) and that the signals are signed integers.

The model is saved as a ".mdl" file to be used in the MATLAB code. With the set of parameters described, the Simulink model is ready for simulations using the MATLAB code given in the next subsection.

**Table 2.2** Details of the blocks in Fig. 2.19

| Block | Library | Configuration |
|---|---|---|
| Constant | Sources | Constant value=input |
| From workspace | Sources | Data=dither |
| Quantizer | Discontinuities | Quantization interval=$M$ |
| Saturation | Discontinuities | Lower limit=n_min$\times M$; upper limit=n_max$\times M$ |
| To workspace | Sinks | Variable name=yout, save format=array |
| Gain | Math operations | |
| Unit delay | Discrete | Initial condition=ic$i$ for accumulator $i$ |
| Sum | Math operations | |

#### 2.6.1.2 MATLAB

In this subsection, we describe the MATLAB code that runs the Simulink model described in the previous subsection and generates the power spectral density of the output of the DDSM. The quantizer step size is $M = 2^{16}$; the input is $\frac{M}{2}$, sim_time $= 2^{18}$, n_min $= -4$ and n_max $= 4$. The output of the DDSM takes on values in the range $\{-4, -3, \cdots, 3, 4\}$. The generated "dither" signal can be selected using the d_sw flag. The dither signal is used in the Simulink model as shown in Fig. 2.19 and it is generated by the following operation:

$$round(rand(1, simtime)).$$

The dither is selected and the initial conditions are set to zero. The Simulink model SQDDSM3rd.mdl is then simulated using the command "sim". After simulation, the average value of the output is calculated as a quick check for correct operation.

The average in this case should be equal to $\frac{\frac{M}{2}}{M} = 0.5$.

```
% Define the step size
n=16;
M=2^n;
% Define the max and min values of the saturation block
n_min=-4;
n_max=4;
% Define the number of simulation points
sim_time=2^18;
% Define the input
input=1*M/(2^1);
% Calculate the variance for linear prediction
variance=1/12;
% Enable (d_sw=1) or disable (d_sw=0) dither
d_sw==1;
if d_sw=1
dither=[1:sim_time;1*round(rand(1,sim_time))]';
else
dither=[1:sim_time;zeros(1,sim_time)]';
end
% Set the initial condition
ic1=0;
ic2=0;
ic3=0;
% Simulate the Simulink model for the third order SQDDSM
sim('SQDDSM3rd.mdl',[1 sim_time]);
% Check the output average
m_y=mean(yout);
% Define a Hanning window
```

```
window=hann(length(yout));
% Calculate the PSD using Periodogram
[Pyy, w] = periodogram(yout,window,sim_time);
% Consider dither and calculate the PSD using the linear prediction
dither_contribution=1/12*1/(M^2);
PSD_predicted=dither_contribution+variance*(2*sin(w/2)).^6;
% Plot output PSD and the PSD using the linear prediction
figure;
semilogx(w/pi,10*log10(pi*Pyy),'b');
hold
semilogx(w/pi,10*log10(PSD_predicted),'k');
grid on
xlabel('Normalized Frequency (xπ rad/sample)','fontsize',14)
ylabel('Power/frequency(dB/rad/sample)','fontsize',14);
set(gca,'fontsize',14)
set(findobj(gca,'Type','line'),'LineWidth',2)
```

The PSD of the output is calculated using the "Periodogram" [7] that calculates:

$$S[k] = \frac{\frac{1}{N_f}\left|\sum_{l=0}^{N_f-1} w[l]x[l]e^{-j\frac{2\pi}{N_f}kl}\right|^2}{\frac{1}{N_f}\sum_{l=0}^{N_f-1}|w[l]|^2},\ 0 \le k < N_f \tag{2.36}$$

where $x[l]$ is the $l$th sample of the signal $x$ with length $N_f$, $w[l]$ is the $l$th sample of the window signal $w$, and $k$ is an integer in the given range. A Hanning window (window=hann(length(yout));) with the same length of the signal is used; it is generated in MATLAB using the command "hann" that calculates the following equation:

$$w[k] = 0.5\left(1 - \cos\left(2\pi\frac{k}{N_w - 1}\right)\right),\ 0 \le k \le N_w - 1 \tag{2.37}$$

where $N_w$ is the length of the window.[13] All the spectral plots in this book use the Periodogram function with a Hanning window, unless otherwise stated.

Considering the dither contribution, the white noise prediction is calculated using

$$S(\omega_k) = \frac{1}{M^2}\frac{1}{12} + \sigma^2\left\{2\sin\left(\frac{\omega_k}{2}\right)\right\}^{2l},\ \omega_k \in \left\{0, \frac{2\pi}{N_f}, \frac{4\pi}{N_f}, \cdots, \pi - \frac{2\pi}{N_f}\right\}, \tag{2.38}$$

where $\sigma^2 = \frac{1}{12}$ and $l$ is the order of the modulator; $l = 3$ in this example.

[13] Here $N_w$ =sim_time.

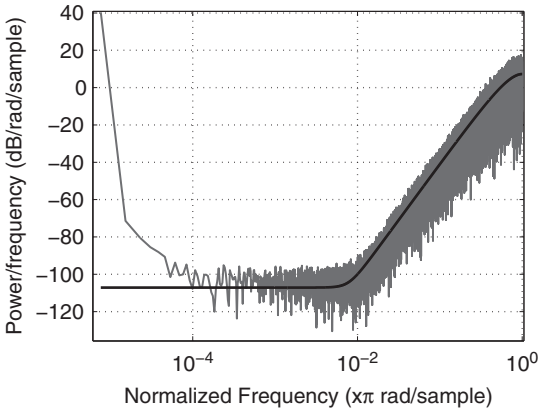



**Fig. 2.20** Simulation result after running the MATLAB code. The DC component corresponds to $X = \frac{M}{2}$

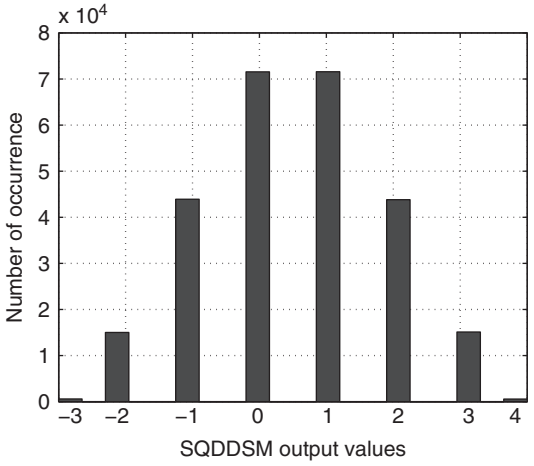



**Fig. 2.21** Distribution of the output samples for the simulated third order SQ-DDSM

Figure 2.20 shows the simulation result after running the prescribed MATLAB code. The noise floor is due to the dither contribution and, as one can see, the quantization noise is pushed away from lower frequencies, as expected.

In Fig. 2.21, we show the distribution of the output samples and in Fig. 2.22 we show some output samples for illustrative purposes. The output in this simulation example occupies 8 levels, as shown in the histogram plot.

## *2.6.2 Multi-Level EFM*

We consider a third order modulator with an all pass STF ($STF(z) = 1$) and an NTF given by Eq. (2.25) with $l = 3$. The Simulink model is shown in Fig. 2.23.

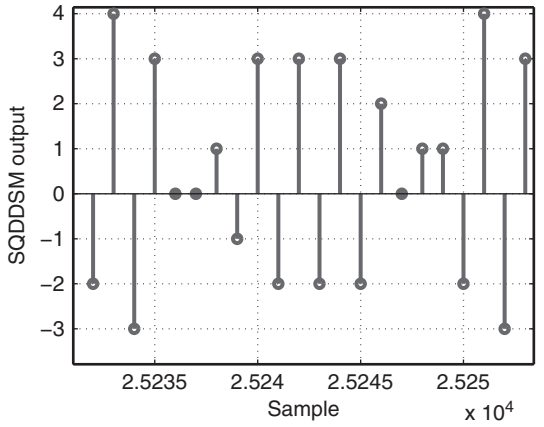



**Fig. 2.22** A portion of the output sequence of the simulated SQ-DDSM

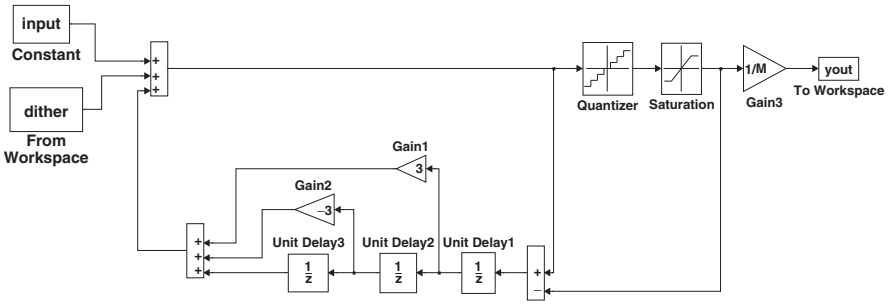



**Fig. 2.23** Simulink model of a third order EFM DDSM (EFM3rd.mdl)

The configuration parameters and the MATLAB code are identical to those of the SQ-DDSM in Section 2.6.1. To calculate the STF and NTF of the modulator, the quantizer and the saturation blocks are replaced by a gain factor ($k = 1$ in this case) and an additive noise source. After simple algebric calculations in the $z$ domain one can determine the STF and NTF.

## *2.6.3 MASH*

In this subsection we consider the MASH DDSM. First we describe the Simulink model and then we explain the MATLAB code.

### 2.6.3.1 Simulink Model

The MASH DDSM architecture we consider in this book is based on the first order error feedback modulator shown in Fig. 2.10a. The Simulink model for the popular

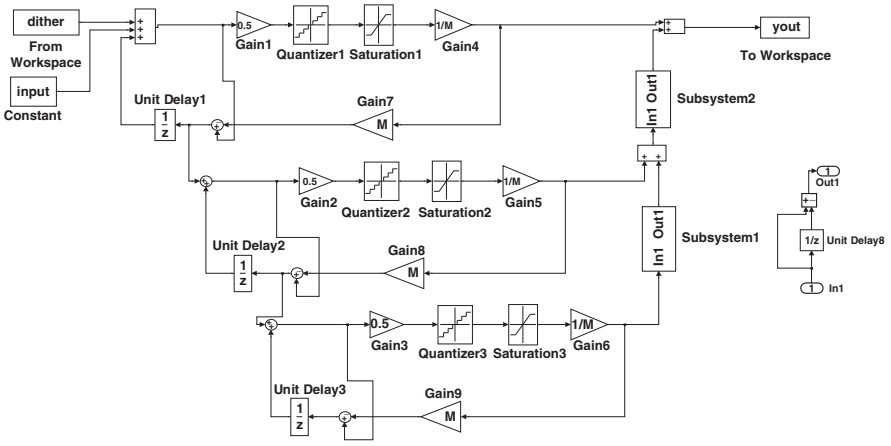



**Fig. 2.24** Block diagram of the simulated third order MASH 1-1-1 DDSM (MASH111.mdl) in Simulink

MASH 1-1-1 DDSM is presented in Fig. 2.24. This DDSM contains three identical EFM1 stages. The first stage comprises Gain1, Quantizer1, Saturation1, Gain4, Gain7, a subtractor, Unit delay1 and the input summing block. The gain block (Gain1) with the value of 0.5, the quantizer block, the saturation block and the gain block of $\frac{1}{M}$ are explicitly used in order to implement the 1-bit quantizer described in Fig. 2.10c. The noise cancellation network comprises Subsystem1, Subsystem2 and two summing blocks. The details of the identical subsystem blocks are shown next to Subsystem1. The configuration parameters of the blocks in the MASH 1-1-1 Simulink model are summarized in Table 2.3.

### 2.6.3.2 MATLAB Code

The MATLAB code is very similar to that presented in Section 2.6.1.2. The output of each EFM1 stage in the MASH modulator is binary (0 or 1). After applying these three 1-bit outputs to the noise cancellation network, the final output of the modulator contains 8 levels in the range $\{-3, -2, \cdots, 3, 4\}$. A representative output

**Table 2.3** Details of the blocks in Fig. 2.24

| Block | Library | Configuration |
|---|---|---|
| Constant | Sources | Constant value=input |
| From workspace | Sources | Data=dither |
| Quantizer | Discontinuities | Quantization interval=$M$ |
| Saturation | Discontinuities | Lower limit=0; upper limit=$M$ |
| To workspace | Sinks | Variable name=yout, save format=array |
| Gain | Math operations | Values shown in Fig. 2.24 |
| Unit delay | Discrete | Initial condition=ic1, ic2 and ic3 for unit delays 1, 2 and 3 respectively |
| Sum | Math operations | |

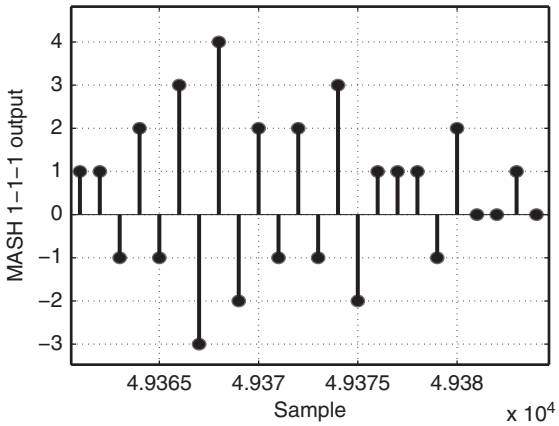



**Fig. 2.25** A portion of the output sequence of the MASH DDSM with the constant input $\frac{M}{2}$ where $M = 2^{16}$

sequence is shown in Fig. 2.25. The average of the output sequence converges to the input divided by $M$ (0.5 in the case that $X = \frac{M}{2}$). The designer needs first to check the average of the output sequence to verify correct operation. The second step is to observe the PSD. When checking the PSD, the designer should measure the slope of the quantization noise spectrum. In the case of the MASH 1-1-1 DDSM, the slope is 60 dB/decade. In Fig. 2.26 we show the output PSD of the MASH 1-1-1 DDSM when dithered with LSB dithering signal at input. As shown in the figure, the slope is 60 dB/decade and there are no obvious spurs in the spectrum.

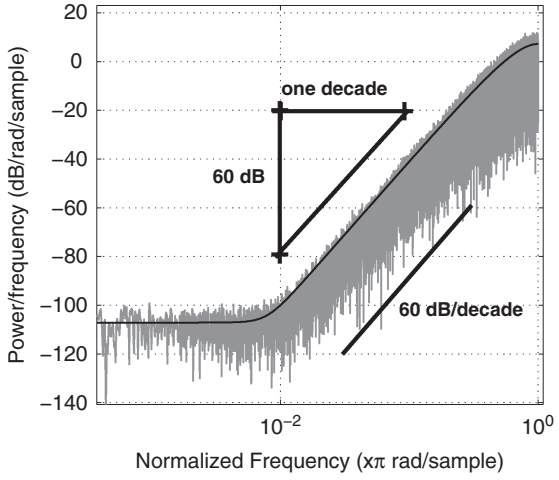



**Fig. 2.26** Output PSD of the simulated MASH 1-1-1 DDSM with dither

## 2.7 Summary

In this chapter, we have reviewed in detail the idea of quantization noise shaping and have described how delta-sigma modulation may be used to implement this concept. In principle, the inband signal-to-quantization noise ratio at the output of a DDSM can be improved by using higher order modulators and by increasing the oversampling ratio.

We classified modulators based on their input signals. Analog modulators belong to two categories: discrete-time and continuous time. They are used primarily in analog-to-digital converters. DDSMs are used in digital-to-analog converters and fractional-N frequency synthesizers. In this book, we are interested primarily in applications where the input to the DDSM is a constant digital word.

We reviewed representative DDSM topologies including single quantizer, error feedback and MASH modulators.

The principles of operation of the delta-sigma DAC were reviewed briefly. Following an explanation of the DAC, the basics of fractional-N frequency synthesizers were presented. We explained the motivation behind the use of a DDSM for implementing fractional frequency division. Through the simulation of a PLL, we showed that the phase noise of a fractional-N synthesizer can be improved if a DDSM is used instead of a simple accumulator for controlling the divider modulus. The idealized DDSM randomizes and noise shapes the divider sequence, thereby removing the fractional tones that appear in the case of an accumulator based synthesizer.

We concluded the chapter by providing Simulink models, MATLAB code and representative simulations for three DDSMs, namely SQ, EFM and MASH.

http://www.springer.com/978-1-4614-0093-6

Minimizing Spurious Tones in Digital Delta-Sigma Modulators

Hosseini, K.; Kennedy, M.P.

2011, XIV, 150 p., Hardcover
ISBN: 978-1-4614-0093-6