NONUNIFORM-TO-UNIFORM DECIMATION FOR DELTA-SIGMA FREQUENCY-TO-DIGITAL CONVERSION

W. Huff I. Galton
UNIVERSITY OF CALIFORNIA, SAN DIEGO
9500 Gilman Dr, La Jolla, CA 92093-0407
whuff@ece.ucsd.edu galton@ece.ucsd.edu

ABSTRACT

Recently, a delta-sigma frequency-to-digital converter (FDC) capable of 14-bit audio bandwidth data conversion at a 10MHz intermediate frequency (IF) was presented. The nonuniform-to-uniform decimation filter, a key component of this FDC, corrects for the nonuniform sampling inherent to the FDC and performs decimation filtering. This paper contains a detailed description of the nonuniform-to-uniform decimation filter and measured performance results for the entire FDC.

1. INTRODUCTION

DIGITAL signal processing is increasingly used in place of analog processing in wireless communication systems to reduce manufacturing costs, improve reliability, and allow programmability, particularly in high-volume, consumer-oriented systems. In receivers, the requisite A/D conversion is usually performed after the radio frequency (RF) signal has been down-converted to an IF in the zero to 100MHz range, and the demodulation is performed digitally. Although conventional ADCs are most commonly used for this purpose, the majority of wireless signal formats are based on frequency or phase modulation. In these cases, an alternative approach is to use a delta-sigma frequency-to-digital converter ($\Delta\Sigma$FDC)—a device that digitizes instantaneous frequency rather than amplitude—instead of an ADC.

Recently, a CMOS delta-sigma phase-locked loop ($\Delta\Sigma$PLL) has been demonstrated [1] that promises to make this an attractive approach. The $\Delta\Sigma$PLL is the key front-end component of a high-performance $\Delta\Sigma$FDC capable of 14-bit audio bandwidth data conversion at a 10MHz IF. To the knowledge of the authors, this level of performance has yet to be achieved using other FDC or bandpass ADC architectures in CMOS circuit technology [2], [3].

A high-level block diagram of the $\Delta\Sigma$PLL-based $\Delta\Sigma$FDC is shown in Figure 1. It consists of a hard limiter, a $\Delta\Sigma$PLL, a digital lowpass filter, a digital nonuniform-to-uniform downsampler, and a conventional digital decimation filter. The theory underlying the $\Delta\Sigma$PLL has been presented in [4], and the implementation details of the CMOS prototype mentioned above have been presented in [1]. The components following the $\Delta\Sigma$PLL in Figure 1 are collectively referred to as the nonuniform-to-uniform decimation filter. The details of the nonuniform-to-uniform decimation filter have not been presented previously and are the focus of this paper.

The $\Delta\Sigma$PLL performs an instantaneous frequency estimate each time that a positive going zero-crossing of the input signal occurs. Thus, the sampling rate of the $\Delta\Sigma$PLL is determined by the input frequency, so a frequency modulated (FM) input signal results in nonuniform sampling intervals.

Many techniques for reconstructing a signal from nonuniformly spaced samples have been described, such as generalized sampling theorems and Lagrange interpolation [5]. However, most of these methods are computationally expensive. Moreover, the quantization noise-shaping inherent to the $\Delta\Sigma$PLL is not generally present in other nonuniform-to-uniform conversion applications. The approach described herein achieves high-precision with very low implementation complexity by combining the nonuniform-to-uniform conversion operation with the decimation filtering required to remove the out-of-band quantization noise. The specific nonuniform-to-uniform decimation filter described is compatible with the $\Delta\Sigma$PLL prototype of [1] and gives rise to the above-mentioned $\Delta\Sigma$FDC performance, but the results are easily generalized to other $\Delta\Sigma$PLL configurations and $\Delta\Sigma$FDC specifications.

2. BACKGROUND

The $\Delta\Sigma$FDC input signal is assumed to have the form $x(t) = A_c \sin\big(\omega_c t + \theta(t)\big)$, where A_c is the amplitude, ω_c is the carrier frequency, and $\theta(t)$ is the *instantaneous phase* of $x(t)$ relative to $\omega_c t$. The instantaneous frequency of $x(t)$ relative to ω_c is $\psi(t) = \frac{d\theta}{dt}$. For the system considered in the paper, it is assumed that $\psi(t)$ is an audio bandwidth signal.

The $\Delta\Sigma$PLL converts the *instantaneous frequency* of its input signal into a digital output sequence with shaped quantization noise in a manner analogous to the way that a $\Delta\Sigma$ modulator converts the *amplitude* of its input signal into a digital output sequence. A functional diagram of the specific $\Delta\Sigma$PLL prototype presented in [1] is shown in Figure 2. It it clocked by an 80 MHz uniform master clock, clk_m, and operates on a hard limited version of the input signal, $x_h(t)$. The $y[n]$ output signal is a quantized, nonuniformly sampled estimate of the instantaneous frequency of $x(t)$ with quantization noise-shaping as will be described shortly. The clk_n output signal is a clock signal wherein each rising edge indicates that new valid data bits are available at the $y[n]$ output.

The internal timing of the $\Delta\Sigma$PLL is controlled by two events: 1) The rising edge of $x_h(t)$, and 2) the rising edge of the "carry" signal. The 4-bit counter is driven by clk_m, and the "carry" signal goes high on its terminal count. In normal operation, the output of the XOR gate goes high when $x_h(t)$ goes high. This switches the positive current source, I_p, on until the "carry" signal goes high at which time it is switched off. The negative

0-7803-4455-3/98/$10.00 © 1998 IEEE

current source, I_n, is switched on for a fixed duration each time the "carry" signal goes high. Two clk_m cycles after the "carry" signal goes high, filtered ADC data is loaded into the counter, effectively setting the time until the next carry. The "carry" signal and its time-shifted derivatives are synchronized to the *actual sample times* of the $\Delta\Sigma$PLL, and the "carry+1" signal is brought out from the $\Delta\Sigma$PLL as clk_n. From the results in [4] it can be shown that the $\Delta\Sigma$PLL of Figure 2, with an appropriate LSB normalization, produces the output sequence

$$y[n] = \frac{4}{\pi}\widehat{\psi}[n] + e_2[n], \tag{1}$$

where $e_2[n]$ is 2nd-order shaped quantization noise with power spectral density (PSD) proportional to $\sin^4(\omega/2)$, and $\widehat{\psi}[n]$ is an estimate of instantaneous frequency corresponding to the change in instantaneous phase between adjacent samples. That is , $\widehat{\psi}[n] = \int_{t_{n-1}}^{t_n} \psi(\tau)d\tau$, where t_n is the time of the n^{th} sample of $x(t)$.

It can also be shown from the theory in [4] that $y[n]$ can be written in terms of the sample times, t_n, as

$$y[n] = 8\left(1 - \frac{\delta_n}{T_c}\right) + e_2[n], \tag{2}$$

where $T_c = 2\pi/\omega_c$ is the nominal period of the input signal, and $\delta_n = t_n - t_{n-1}$.

The average sample-rate of the $\Delta\Sigma$PLL is equal to the center frequency of $x(t)$ which can range between 8.9MHz and 10MHz, but the sampled signal, $\psi(t)$, is an audio-bandwidth signal. Therefore, $\widehat{\psi}[n]$ is oversampled by a factor of approximately 200, and resides at very low frequencies in the discrete-time spectrum. In contrast, due to its $\sin^4(\omega/2)$ PSD shape, most of the quantization noise power resides at high frequencies.

The $\Delta\Sigma$PLL is similar to a $\Delta\Sigma$ modulator in that both quantize an oversampled version of the input signal such that the quantization noise is spectrally shaped to reside at high frequencies. However, an important difference is that the nonuniform sampling of the $\Delta\Sigma$PLL causes a slight spectral spreading of the oversampled signal. As discussed below, this spreading effect places a design constraint on the passband flatness of the subsequent filter stage. The nonuniform sampling and resulting spreading effect are corrected for by the nonuniform-to-uniform decimation filter.

3. DECIMATION ARCHITECTURE

The nonuniform-to-uniform decimation technique takes advantage of the property that the output sequence from the $\Delta\Sigma$PLL can be used to estimate both δ_n and $\widehat{\psi}[n]$. From the sequence of δ_n values, the actual sample times can be estimated, and from the sequence $\widehat{\psi}[n]$, frequency estimates for *desired sample times* can be calculated by 2 point interpolation about actual sample times.

3.1 Comb Filtering

The $\Delta\Sigma$PLL output sequence, $y[n]$, is filtered by $H(z)$. For this implementation, $H(z)$ is a $comb^2$ filter with a transfer function $H(z) = \left[\frac{(1-z^{-4})}{(1-z^{-1})}\right]^2$. Its purpose is to remove as much of the out-of-band quantization noise as practical without significantly distorting the components of the oversampled signal that underwent spectral spreading as a result of the nonuniform sampling. Thus, passband flatness and stopband attenuation are both design constraints. In addition, since the discrete-time frequency of the desired stopband shifts with instantaneous carrier frequency, a wide stopband is desired. The particular choice of $H(z)$ represents a good compromise with respect to these tradeoffs. Moreover, it can be implemented efficiently using a multiplier-free recursive structure.

The filtered sequence, $y_f[n]$, still contains the low-frequency signal information, but with much of the high-frequency quantization noise removed [6]. Thus, from (1), (2), and the frequency response implied by $H(z)$,

$$y_f[n] = \frac{64}{\pi}\widehat{\psi}[n] + e_f[n], \tag{3}$$

and

$$y_f[n] = 128\left(1 - \frac{\delta_n}{T_c}\right) + e_f[n], \tag{4}$$

where $e_f[n]$ is the residual quantization noise remaining after $e_2[n]$ is filtered by $H(z)$. The passband droop associated with $H(z)$ has been neglected.

3.2 Nonuniform-to-Uniform Downsampler

The two sub-blocks comprising the nonuniform-to-uniform downsampler of Figure 1 are shown in detail in Figure 3. The *uniform clock generator* extracts the sample time information from $y_f[n]$ to determine which pairs of actual sample times straddle the desired sample times. The *interpolating downsampler* performs frequency estimation at the desired sample times by two-point interpolation about these actual sample times. The desired sample times are defined to occur at uniform intervals of T_s, where $T_s = PT_c$, and P is the average downsampling ratio. For the implementation described in this paper, $P = 4$.

From Figure 3 and equation (4), it follows that

$$r[n] = -128\left(\frac{\delta_n}{T_c}\right) + 256 + e_f[n]. \tag{5}$$

The purpose of the register $t[n]$ shown in Figure 3 is to accumulate time interval estimates. During normal operation, $t[n]$ stores the value of the actual time estimate relative to the desired sample time. This value can be either positive or negative. If $t[n]$ is initially negative, it can be seen from Figure 3 that the scaled output of the hard limiter (clk_u) is -256, which cancels the $+256$ DC offset of $r[n]$ from (5), and causes a value proportional to δ_n to be added to $t[n-1]$ thereby obtaining $t[n]$. Since δ_n is always positive by definition, $t[n]$ will monotonically increase while clk_u is -1. After $t[n]$ exceeds zero, the output of the hard limiter switches from -1 to +1, so the scaled hard limiter output will combine with the +256 DC offset of $r[n]$, insuring that the next

value of $t[n]$ is negative. This effectively sets a new desired sample time. Thus, each rising edge of clk_u occurs when the accumulated time estimate first exceeds each desired sample time. At this time, from (5) and Figure 3 the relation between the actual time estimate and the desired sample time is

$$t[n] = \frac{128}{T_c}(t_n - T_k) + e_t[n], \tag{6}$$

where t_n is the actual sample time, T_k is the desired sample time, and $e_t[n]$ is the quantization noise remaining at $t[n]$.

The interpolating downsampler processes the information contained in $y[n]$ to extract uniformly spaced estimates of the instantaneous frequency of $x(t)$. The first processing stage in the interpolating downsampler accumulates $y_f[n]$ to obtain $\tilde{\theta}[n]$. The n^{th} sample of the resulting sequence is the *nonuniform phase estimate* at time t_n. Interpreting the LSB value to be $\pi/64$, the sequence of nonuniform phase estimates can be written as

$$\tilde{\theta}[n] = \frac{\pi}{64}\sum_{k=1}^{n} y_f[k] + \tilde{\theta}[0]. \tag{7}$$

where $\tilde{\theta}[0]$ is an arbitrary initial value.

As described above, the rising edge of the uniform clock clk_u occurs when the actual sample time first exceeds the next desired sample time, T_k. The actual sample times t_{n-1} and t_n thus straddle the desired sample time. From the nonuniform phase estimates at t_{n-1} and t_n, two-point interpolation is used to find the *uniform phase estimate*, $\tilde{\theta}_u[k]$, at time T_k. This is similar to the two-point interpolation method introduced in [7], in which it was shown that the interpolation error is small provided the modulation frequency is much smaller than the carrier frequency, as is the case here.

The two-point interpolation expression is

$$\tilde{\theta}_u[k] = \tilde{\theta}[n] - \Big(\tilde{\theta}[n] - \tilde{\theta}[n-1]\Big)\left(\frac{t_n - T_k}{t_n - t_{n-1}}\right), \tag{8}$$

which is depicted graphically in Figure 4. An expression equivalent to (8) can be derived from Figure 3, the definition for δ_n, and equations (4), (6), and (7). Neglecting the LSB normalization in (7), and neglecting quantization noise,

$$\tilde{\theta}_u[k] = \tilde{\theta}[n] - \frac{t[n]}{128}\left(\frac{y_f[n]}{1 - \frac{1}{128}y_f[n]}\right). \tag{9}$$

The system of Figure 3 implements this form of the two-point interpolation expression.

It follows from (9) that one multiply and one divide are required for every set of $P = 4$ $\Delta\Sigma$PLL output samples. For the specific implementation described in this paper, it follows that the multiplication and division rates are no greater than 2.5MHz each. Moreover, the numbers of bits used for $y[n]$ and $t[n]$ are low (6 and 10, respectively). Consequently, the computational requirements of the interpolation are not excessive.

As seen in Figure 3, the $1-z^{-1}$ operation performed upon $\tilde{\theta}_u[k]$ produces *uniform frequency estimates*, $\widehat{\psi}_u[k]$. Ignoring quantization noise and interpolation error, it follows that $\widehat{\psi}_u[k] = \tilde{\theta}_u[k] - \tilde{\theta}_u[k-1]$, which is proportional to the phase difference between times T_{k-1} and T_k. Thus, the sequence $\widehat{\psi}_u[k]$ corresponds to frequency estimates at uniformly spaced sampling intervals.

3.3 Conventional Decimation Filter

The final processing block shown in Figure 1 is a conventional 50-fold lowpass decimation and equalization filter. It removes the remaining out-of-band quantization noise and reduces the output sequence to the final desired sampling rate. Equalization filtering is incorporated in the block to compensate for the passband droop of $H(z)$ as well as that imposed by the effective moving average of the desired signal performed by the nonuniform-to-uniform conversion processing.

4. EXPERIMENTAL RESULTS

The performance of the $\Delta\Sigma$FDC with nonuniform-to-uniform decimation was measured by combining the existing $\Delta\Sigma$PLL prototype [1] with a software implementation of the nonuniform-to-uniform decimation filter. The bus widths and other processing details were implemented exactly as shown in Figures 1 and 3, so identical results would be expected from a fully integrated VLSI implementation.

The signal to noise and distortion ratio (SINAD) and spurious free dynamic range (SFDR) measurements were made using an input signal with a sinusoidal instantaneous frequency function: $\psi(t) = 2\pi\Delta F \sin(2\pi f_1 t)$. Thus $x(t)$ was a sinusoidally modulated FM signal with a peak frequency deviation of ΔF and a modulation frequency of f_1. Measurements were taken with numerous values of ΔF and f_1. The worst case results for peak SINAD and peak SFDR were 85dB and 88dB, respectively, indicating 14b worst-case performance.

Representative examples of the measured data are shown in Figures 5 and 6. Figure 5 shows the PSD of $\widehat{\psi}_d[m]$ corresponding to $\Delta F = 1$MHz and $f_1 = 8$kHz. The measured (non-peak) SINAD and SFDR for this example are 85dB and 88dB, respectively. Figure 6 shows measured values of SINAD and SFDR as a function of ΔF (recall that ΔF is to the $\Delta\Sigma$FDC what amplitude is to a $\Delta\Sigma$ADC) again for $f_1 = 8$kHz. In this case, the measured peak SINAD and peak SFDR are 89dB and 94dB.

An intermodulation measurement was made using an input signal with a two-tone instantaneous frequency function of the form: $\psi(t) = 2\pi\Delta F_1 \sin(2\pi f_1 t) + 2\pi\Delta F_2 \sin(2\pi f_2 t)$. Figure 7 shows data corresponding to the PSD of $\widehat{\psi}_d[m]$ with $\Delta F_1 = \Delta F_2 = 250$kHz, $f_1 = 22.1$kHz, and $f_2 = 24.3$kHz. As shown in the figure, the maximum in-band intermodulation (IM) product for this test was -87dB.

REFERENCES

1. I. Galton, W. Huff, P. Carbone, E. Siragusa, "A delta-sigma PLL for 14b 50kSample/s frequency-to-digital conversion of a 10 MHz FM signal," *Dig. of Tech. Papers, IEEE Inter. Solid-State Circuits Conf,* vol 41, pp. 366-7, Feb. 1998.
2. M. Hovin, A. Olsen, T. S. Lande, C. Toumazou, "Delta-sigma modulators using frequency-modulated intermediate values," *IEEE J. of Solid-State Circuits,* vol. 32, no. 1, pp. 13-22, Jan. 1997.
3. M. Song, J. Park, W. Joe, M. J. Choe, B. S. Song, "A fully-integrated 5MHz-IF FM demodulator," *Proc. of the IEEE Custom Integrated Circuits Conference*, pp. 525-8, May, 1997.
4. I. Galton, "Analog-input digital phase-locked loops for precise frequency and phase demodulation," *IEEE Trans. on Circuits and Syst.–II: Analog and Digital Sig. Proc.*, vol. 42, no. 10, pp. 621-630, Oct 1995.
5. A.J. Jerri, "The Shannon sampling theorem - its various extensions and applications: a tutorial review," *Proc. of the IEEE*, vol. 65, no. 11, pp. 1565-96, Nov, 1977.
6. J.C. Candy, "Decimation for sigma delta modulation," *IEEE Trans. on Communications*, vol.COM-34, (no.1), pp. 72-6 Jan. 1986.
7. S.K. Ray, "Zero-crossing-based approximate demodulation of wide-deviation FM," *IEE Proc.*, vol. 131, part F, no. 1, pp. 47-51, Feb 1984.

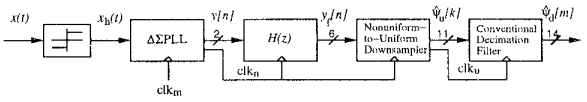

Figure 1: ΔΣ frequency-to-digital converter (ΔΣFDC)

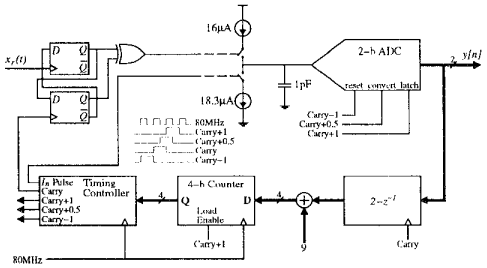

Figure 2: ΔΣPLL prototype functional diagram.

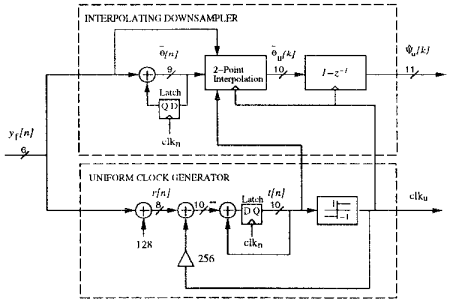

Figure 3: Nonuniform-to-uniform downsampler.

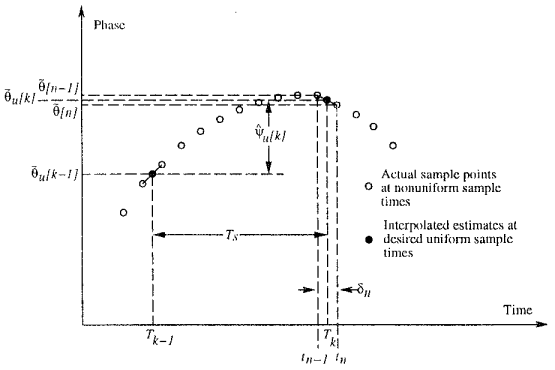

Figure 4: Interpolation between actual sample points to find desired sample points

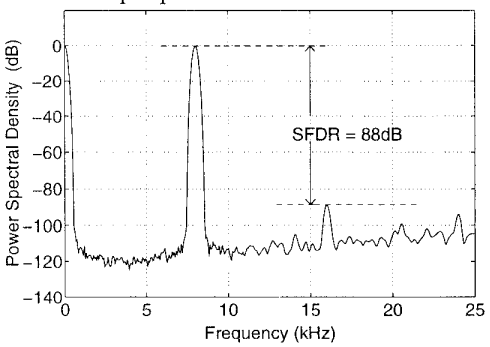

Figure 5: Measured PSD for single tone test

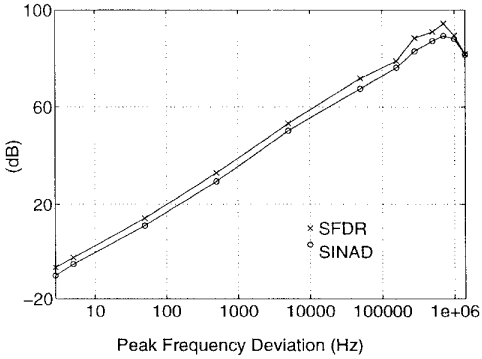

Figure 6: Measured SINAD and SFDR vs. peak frequency deviation

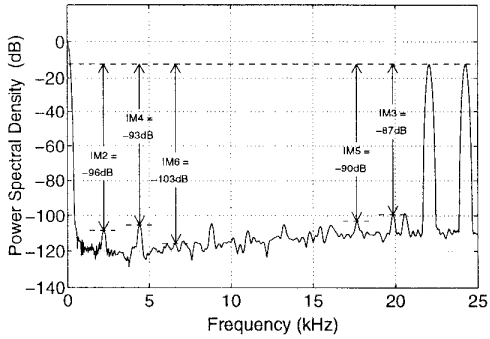

Figure 7: Measured PSD for two tone test