A Delta–Sigma PLL for 14-b, 50 kSample/s Frequency-to-Digital Conversion of a 10 MHz FM Signal

Ian Galton, William Huff, Paolo Carbone, and Eric Siragusa

Abstract— **In many wireless applications, it is necessary to demodulate and digitize frequency or phase modulated signals. Most commonly, this is done using separate frequency discrimination and analog-to-digital (A/D) conversion. In low-cost IC technologies, such as CMOS, precise analog frequency discrimination is not practical, so the A/D conversion is usually performed in quadrature or at a nonzero intermediate frequency (IF) with digital frequency discrimination. While practical, the approach tends to require complicated A/D converters, and accuracy is usually limited by the quality of the A/D conversion. This paper presents an alternative structure, referred to as a delta–sigma frequency-to-digital converter ($\Delta\Sigma$FDC), that simultaneously performs frequency demodulation and digitization. The $\Delta\Sigma$FDC is shown to offer high-precision performance with very low analog complexity. A prototype of the key component of the $\Delta\Sigma$FDC has been fabricated in a 0.6 μm, single-poly, CMOS process. The prototype achieved 50 kSample/s frequency-to-digital conversion of a 10 MHz frequency-modulated signal with a worst case signal-to-noise-and-distortion ratio of 85 dB and a worst case spurious-free dynamic range of 88 dB.**

I. Introduction

DIGITAL signal processing is increasingly used in place of analog processing in wireless communication systems to reduce manufacturing costs, improve reliability, and allow computer access. In receivers, the requisite digitization usually is performed after the radio-frequency (RF) signal has been down-converted to an intermediate frequency (IF) in the 0–100 MHz range. The majority of wireless signal formats are based on frequency or phase modulation, so the demodulation process usually involves some form of frequency discrimination.

Conventional IF receiver architectures that perform frequency demodulation and digitization are shown in Fig. 1. Each performs frequency demodulation and digitization and therefore can be viewed as a type of *frequency-to-digital converter* (FDC). The conceptually simplest system consists of an analog frequency discriminator and an analog-to-digital converter (ADC), as shown in Fig. 1(a). While practical in discrete-component systems, high-precision analog frequency discrimination is difficult to achieve in low-cost integrated circuit (IC) technologies such as CMOS. The systems shown in Fig. 1(b) and (c) avoid this difficulty through the use of digital discriminators. The system of Fig. 1(b) performs in-phase and quadrature demodulation, dual A/D conversion, and digital discrimination, and the system of Fig. 1(c) performs bandpass A/D conversion and digital discrimination. Recent advances in bandpass $\Delta\Sigma$ADC's make the system of Fig. 1(c) particularly attractive for IC implementation because it avoids the added complexity of analog quadrature downconversion and dual A/D conversion [1]. Nevertheless, the implementation of high-precision integrated ADC's, particularly bandpass ADC's, is a significant design challenge, and their implementation complexity is significant. Furthermore, in CMOS technology, bandpass ADC's are usually limited to about 12 bits of precision [2]–[4].

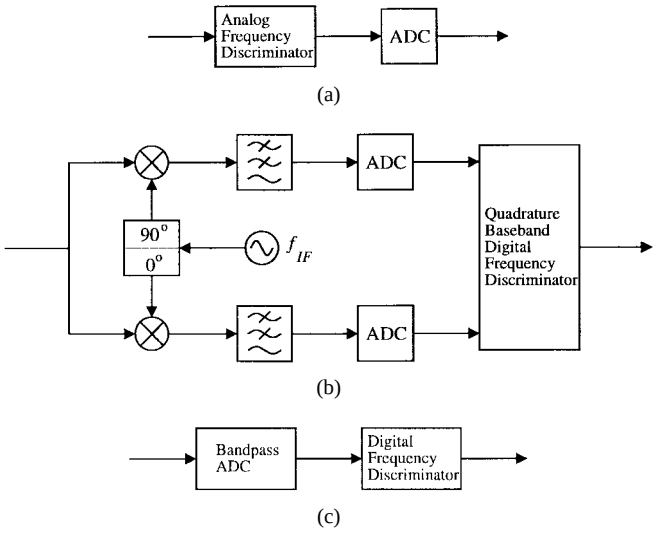


Fig. 1. Common systems that use separate A/D conversion and frequency discrimination to achieve the equivalent of frequency-to-digital conversion.

Alternatively, discrete-time phase-locked loops (DPLL's) can be used to perform frequency-to-digital (F/D) conversion [5]. Most of the known DPLL structures involve only coarse A/D conversion and tend to have much lower analog circuit complexity than the systems of Fig. 1. However, in conventional DPLL's, much of the error introduced by the coarse quantization resides within the signal band, which significantly limits their performance as FDC's [6]–[8].

Manuscript received April 30, 1998; revised July 28, 1998. This work was supported by the National Science Foundation and by the California MICRO program in conjunction with Rockwell International Corp.

I. Galton, W. Huff, and E. Siragusa are with the Department of Electrical and Computer Engineering, University of California, San Diego, La Jolla, CA 92093 USA.

P. Carbone is with the Instituto di Elettronica, Universitá di Perugia, Perugia 06125 Italy.

Publisher Item Identifier S 0018-9200(98)08866-0.

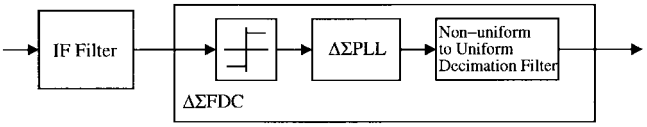


Fig. 2. A high-level view of the prototype FDC preceded by an IF filter.

Recently, new DPLL-like structures have been proposed that use oversampling and quantization noise shaping to limit the portion of the quantization noise that resides within the signal band [9]–[12]. The key component of the FDC presented in this paper, referred to as a delta–sigma phase-locked loop ($\Delta\Sigma$PLL), is one of these DPLL-like structures [10]. The resulting FDC achieves true 14 bit audio bandwidth data conversion at a 10 MHz IF with very low analog and moderate digital circuit complexity [13]. 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.

II. Overview of the $\Delta\Sigma$FDC

A general frequency-modulated (FM) signal at a carrier frequency of $w_{\rm IF}$ has the form

$$x_r(t) = A\sin\left(\omega_{\rm IF}t + \int_{t_0}^{t}\psi(\tau)\,d\tau + \theta_0\right) \tag{1}$$

where A is a constant amplitude, $\psi(t)$ is the *instantaneous frequency* relative to $\omega_{\rm IF}$, and θ_0 is an arbitrary initial phase of the signal at time t_0. Given $x_r(t)$ as an input signal, F/D conversion is the process of extracting, sampling, and digitizing $\psi(t)$.

The prototype FDC is referred to as a $\Delta\Sigma$FDC. As shown in Fig. 2, the $\Delta\Sigma$FDC consists of a hard limiter, a $\Delta\Sigma$PLL, and a nonuniform-to-uniform (NTU) decimation filter. This section describes the operation of these components at the behavioral level of Fig. 2. The implementation details of the components are deferred to Sections III and IV.

The $\Delta\Sigma$PLL operates on a hard limited version of the FM input signal and generates a 2-b output sequence given by

$$y[n] = \frac{4}{\pi}\hat{\psi}[n] + e[n] \tag{2}$$

where $\hat{\psi}[n]$ is a sampled estimate of the instantaneous frequency $\psi(t)$ and $e[n]$ is spectrally shaped quantization noise equivalent to that of a conventional second-order $\Delta\Sigma$ modulator [14]. Thus, $e[n]$ has a power spectral density (PSD) given by

$$S_{ee}(e^{j\omega}) = \frac{4\Delta^2}{3}\sin^4(\omega/2). \tag{3}$$

It follows that, from a signal-processing point of view, the $\Delta\Sigma$PLL is identical to a second-order $\Delta\Sigma$ modulator with input sequence $(4/\pi)\hat{\psi}[n]$. For the $\Delta\Sigma$PLL prototype, the output sample rate is approximately 10 MHz, and $\psi(t)$ is taken to have a bandwidth of 25 kHz. Therefore, the effective oversampling ratio is approximately 200.

As in a $\Delta\Sigma$ modulator, the oversampling causes the desired signal $\hat{\psi}[n]$ to reside at low frequencies, and noise shaping causes the quantization noise $e[n]$ to reside primarily at high frequencies in the discrete-time spectrum. Consequently, much of the quantization noise can be removed by a low-pass digital decimation filter following the $\Delta\Sigma$PLL. However, a complication arises in the case of the $\Delta\Sigma$PLL because it samples the instantaneous frequency at nonuniformly spaced times. This has two effects: it causes the bandwidth of $\hat{\psi}[n]$ to be slightly larger than had it been sampled uniformly and it gives rise to harmonic distortion. The slight increase in the sampled signal bandwidth is not a significant problem in that it simply reduces the effective oversampling ratio by a small amount. However, for applications such as broadcast FM, the distortion caused by nonuniform sampling would be intolerable if left uncorrected.

Fortunately, the necessary correction can be performed in conjunction with the decimation filtering at the cost of a relatively small amount of additional digital hardware [15]. The resulting structure is referred to as the NTU decimation filter. A hardware-efficient version of the structure is presented at the register-transfer level in Section IV. Through a combination of digital filtering, nonuniform downsampling, and two-point interpolation, the NTU decimation filter is able simultaneously to remove most of the out-of-band quantization noise and correct the distortion to better than 14-b linearity.

As demonstrated by the results of this paper, the primary advantage of the $\Delta\Sigma$FDC over the F/D conversion methods shown in Fig. 1 is that it achieves excellent performance with low analog complexity. Because of the hard limiter, the $\Delta\Sigma$FDC belongs to the class of FM demodulators referred to as limiter-discriminators. Such discriminators are widely used in commercial receivers [16]. Although they tend to be more susceptible to out-of-band interferers than the systems of Fig. 1(b) and (c), the availability of low-cost ceramic, crystal, and surface acoustic wave filters centered at various convenient IF frequencies including 10 MHz makes them viable in a large variety of applications. Nevertheless, in applications where the IF filters are unable to reduce interferers to at least several decibels below the desired signal, the limiting operation represents a disadvantage of the $\Delta\Sigma$FDC relative to the systems of Fig. 1(b) and (c).

III. The $\Delta\Sigma$PLL

A. Architecture

The functional diagram of the $\Delta\Sigma$PLL prototype is shown in Fig. 3. It consists of a charge pump and associated gating logic, a 2-b ADC, a $2 - z^{-1}$ block, a digital constant adder, a 4-b counter, and a timing controller. Two events control the timing: 1) transition of $x_h(t)$ from low to high and 2) transition of the "Carry" signal from low to high. The 4-b counter is driven by an 80 MHz external clock, 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 causes the positive current source I_p to be connected to the capacitor. The current source remains connected to the capacitor until the "Carry" signal goes high, at which time it is disconnected from

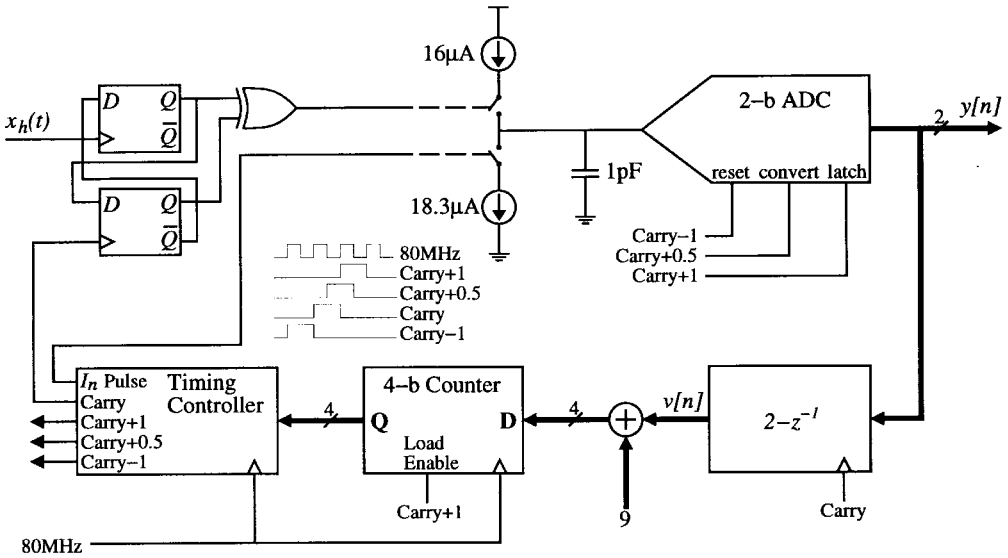


Fig. 3. The $\Delta\Sigma$PLL functional diagram.

the capacitor. The negative current source I_n is connected to the capacitor for a fixed duration each time the "Carry" signal goes high. Two 80 MHz clock cycles after the "Carry" signal goes high, the counter is loaded with a number obtained by filtering the ADC output by $2 - z^{-1}$ and adding nine. This effectively sets the time until the next "Carry," which closes the feedback loop.

Because it is generated by the 4-b counter, the "Carry" signal and its time-shifted derivatives are synchronous with the 80 MHz clock. The 2-b ADC samples and quantizes on the rising edges of the "Carry + 0.5" signal, and the data are made available at the output of the $\Delta\Sigma$PLL on the rising edges of the "Carry + 1" signal.

B. Signal-Processing Details

It will now be shown that, from a signal-processing point of view, the $\Delta\Sigma$PLL is equivalent to a second-order $\Delta\Sigma$ operating on a sampled version of the instantaneous frequency, $(4/\pi)\hat{\psi}[n]$, as claimed in Section II. The practical benefit of quantitatively establishing this connection is that it allows application of the well-established body of knowledge regarding second-order $\Delta\Sigma$ modulator performance to the $\Delta\Sigma$PLL. In particular, the equivalence implies that the $\Delta\Sigma$PLL quantization noise characteristics (e.g., spectral shape, tonal qualities, etc.) are identical to those of the second-order $\Delta\Sigma$ modulator [10].

The starting point is the simplified but functionally equivalent version of the $\Delta\Sigma$PLL shown in Fig. 4(a). The current sources and capacitor have the same values as those shown in Fig. 3. The edge-activated set-reset latch corresponds to the two flip-flops and the XOR gate in Fig. 3, and the *pulse stretcher* is a digital one-shot implemented as part of the timing controller in Fig. 3. The only difference between the two versions of the $\Delta\Sigma$PLL is that $9 + v[n]$ is loaded into the 4-b counter each time the "Carry + 1" signal goes high in the version of Fig. 3, whereas $8 + v[n]$ is loaded into the 4-b counter each time the "Carry" signal goes high in the version of Fig. 4(a). It is easy to verify that this difference does not alter the "Carry" signal or any of its time-shifted derivatives, and therefore has no effect on $y[n]$. Consequently, the properties derived in the following for the simplified $\Delta\Sigma$PLL version of Fig. 4(a) also hold for the implemented $\Delta\Sigma$PLL version of Fig. 3.

Fig. 4(b) shows how signals flow through the $\Delta\Sigma$PLL. At the input to the system, the signal information is represented by the sequence of times, $t_n, n = 1, 2, \cdots$, of the rising edges of $x_h(t)$. Similarly, the feedback information is represented by the sequence of times, $\tau_n, n = 1, 2, \cdots$, of the rising edges of the "Carry" signal. The time information controls the gating of the current sources into the summing capacitor and therefore gets converted into a sequence of analog voltages across the summing capacitor. The ADC converts the sequence of analog voltages to a sequence of digital values. The digital sequence ultimately controls the data that gets loaded into the counter, which generates the feedback signal τ_n.

The input signal $x_h(t)$ changes the $\Delta\Sigma$PLL state only at the discrete sequence of times t_n. Therefore, t_n can be viewed as the nth *sample time* of the input signal. Accordingly, the nth *sample period* will be defined as $\delta_n = t_n - t_{n-1}$. Given that $x_h(t)$ corresponds to a hard limited FM signal, it follows that the sample period varies as a function of n, as mentioned in Section II.

In the following, the signal processing performed by the $\Delta\Sigma$PLL is explained by gradually replacing the components shown in Fig. 3 with generic signal-processing components. The final result is an equivalent system that operates directly on $\hat{\psi}[n]$ and has the form of a second-order $\Delta\Sigma$ modulator.

It is first shown that the set-reset latch, the pulse stretcher, and the charge pump together perform the signal-processing operations of a $\tau_n - t_n$ differencer, a constant offset, and a discrete-time integrator, as indicated in Fig. 5. The heuristics behind this assertion are as follows. In each sample period, the positive current source is connected to the summing capacitor for $\tau_n - t_n$ seconds, and the negative current source is connected to the capacitor for a fixed fraction of the sample

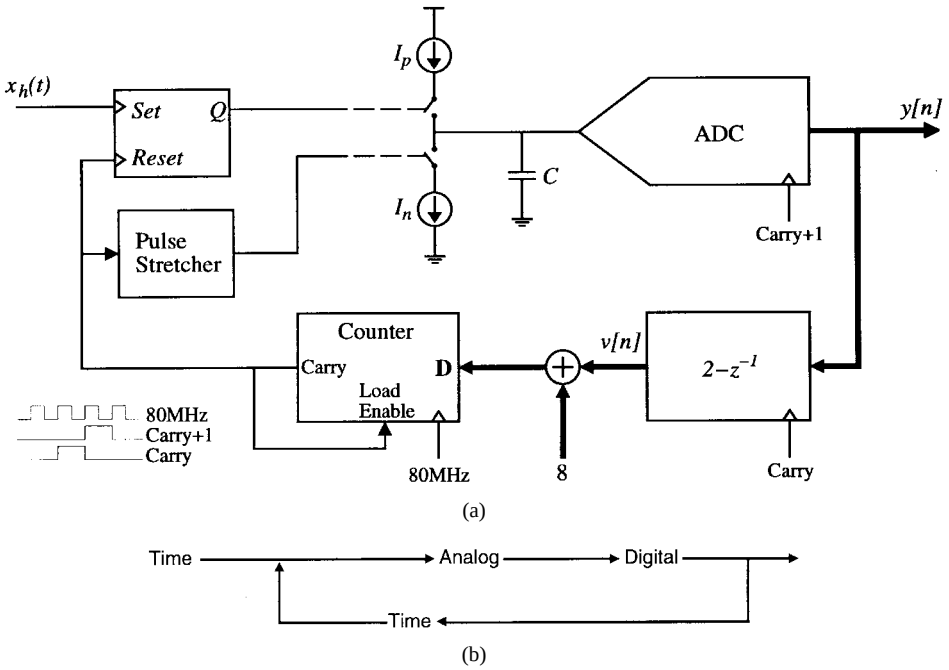


Fig. 4. (a) A functionally equivalent but simplified version of the $\Delta\Sigma$PLL. (b) The flow of signals in the $\Delta\Sigma$PLL.

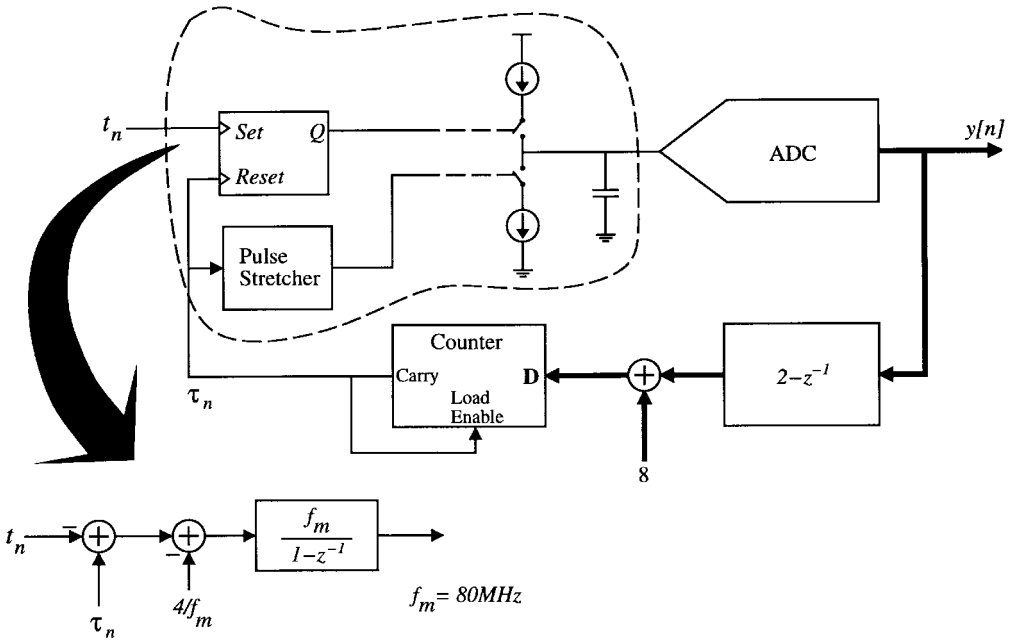


Fig. 5. The signal-processing operations performed by the charge pump and associated logic.

period, namely, 3.5 periods of the 80 MHz external clock. Thus, in each sample period, a "packet" of charge proportional to $\tau_n - t_n$ minus a constant is added to the capacitor. The capacitor is never reset, so it simply accumulates these packets of charge, thereby performing the function of a discrete-time integrator.

Specifically, by this reasoning, the voltage across the capacitor at time τ_n can be written as

$$v_c(\tau_n) = v_c(\tau_{n-1}) - \frac{3.5I_n}{f_m C} + \frac{(\tau_n - t_n)I_p}{C}.$$

The value of the positive current source is nominally $I_p = f_m C\Delta$, where Δ is the step size of the ADC and the ratio of the positive and negative currents is nominally $I_p/I_n = 3.5/4$. Hence, in units of V/Δ, the value at the input to the ADC at time τ_n is given by

$$v_c[n] = v_c[n-1] + f_m[(\tau_n - t_n) - 4/f_m].$$

The $\tau_n - t_n$ differencer, constant offset, and discrete-time integrator shown in Fig. 5 implement this difference equation. Deviations of I_p, C, and Δ from their ideal values result in an integrator gain error, and a deviation in the ratio I_p/I_n from

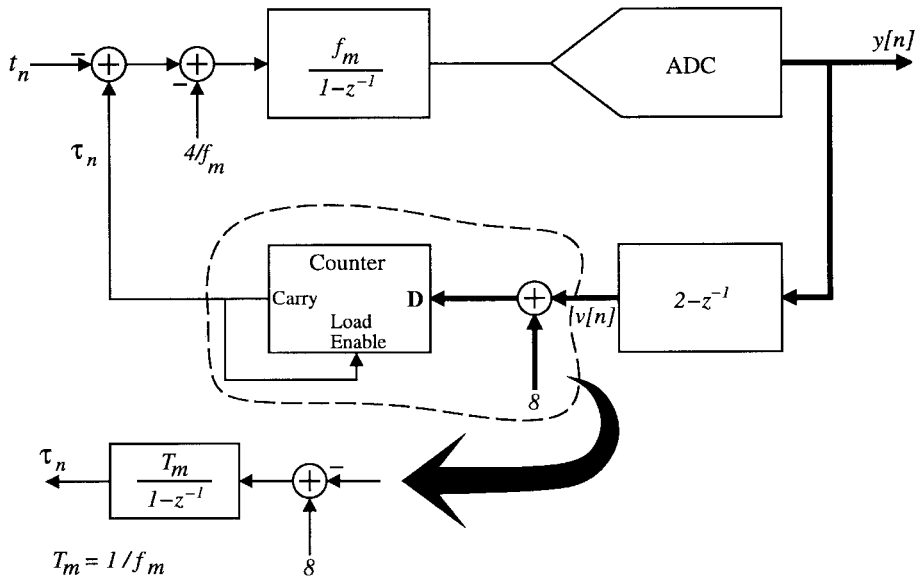


Fig. 6. The signal-processing operations performed by the counter.

its nominal value of 3.5/4 results in an offset error prior to the integrator. The effects of such deviations with respect to $\Delta\Sigma$PLL performance are considered shortly.

It is next shown that the 4-b counter performs the signal-processing operation of another discrete-time integrator. The heuristics behind this assertion are as follows. Recall that the data sequence generated by the counter is *defined* to be the sequence of absolute times τ_n of the rising edges of the "Carry" signal. The "Carry" signal goes high when the counter reaches its terminal count of 15, and the "D" input is latched into the counter one 80 MHz clock period after each rising edge of the "Carry" signal. Thus, the value loaded into the counter during the nth sample period affects not only τ_n but also all future values of the sequence, namely, $\tau_{n+1}, \tau_{n+2}, \cdots$. For example, suppose that a value of nine is loaded into the counter during the nth sample period. Then the "Carry" signal goes high $1 + 15 - 9 = 7$ periods of the 80 MHz external clock after its previous rising edge, so $\tau_n = \tau_{n-1} + 7T_m$, where $T_m = 1/f_m$ is one 80 MHz clock period. If a value of six is loaded into the counter during the next clock period, the "Carry" signal goes high again after $1 + 15 - 6 = 9$ periods of the 80 MHz clock, so $\tau_{n+1} = \tau_{n-1} + 7T_m + 9T_m$. In this fashion, the values loaded into the counter are discrete-time integrated.

Generalizing from this example gives

$$\begin{aligned}\tau_n &= \tau_{n-1} + T_m + 15T_m - T_m(v[n] + 8) \\ &= \tau_{n-1} + T_m(8 - v[n]).\end{aligned}$$

Therefore, the $8 + v[n]$ operation and the counter in the $\Delta\Sigma$PLL can be replaced by an $8 - v[n]$ operation and a discrete-time integrator, as indicated in Fig. 6. Note that Fig. 6 also contains the replacement indicated in Fig. 5.

Fig. 7 shows the $\Delta\Sigma$PLL with the replacements indicated in Figs. 5 and 6 and two additional replacements. First, the ADC has been replaced by its signal-processing equivalent of a unit delay (corresponding to an actual delay of δ_n) followed by a 2-b quantizer with step-size Δ. Second, the input signal

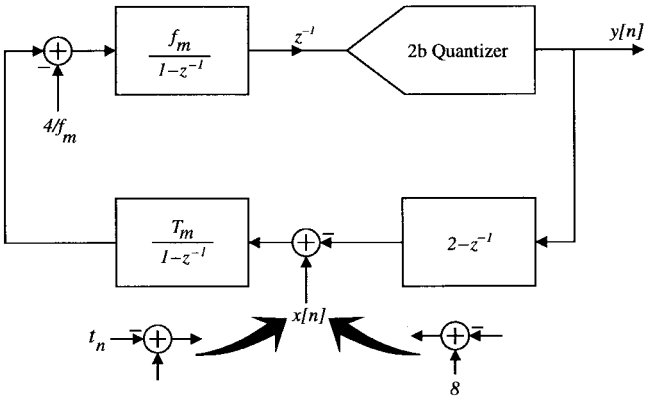


Fig. 7. The signal-processing equivalent of the $\Delta\Sigma$PLL.

t_n and the addition of eight have been combined into a single equivalent input $x[n]$, given by

$$x[n] = 8\left(1 - \frac{t_n - t_{n-1}}{8T_m}\right). \tag{4}$$

It is straightforward to verify that the system of Fig. 7 can be redrawn in the well-known form of a second-order $\Delta\Sigma$ modulator, as shown in Fig. 8. The only unusual aspect of the $\Delta\Sigma$ modulator shown in Fig. 8 is the addition of four prior to the second integrator. However, this addition has no effect on the performance of the system because the transfer function between it and the output has a zero at dc.

To complete the analysis, it remains only to establish the relationship between $x[n]$ as given by (4) and the instantaneous frequency $\psi(t)$. From (1), the phase of $x_r(t)$ can be written as

$$p_x(t) = \omega_{\text{IF}} t + \int_{t_0}^{t} \psi(\tau)\, d\tau + \theta_0. \tag{5}$$

The nth positive-going zero crossing of $x_r(t)$ occurs at time t_n, so it follows that $p_x(t_n) - p_x(t_{n-1}) = 2\pi$. With (5) and $\omega_{\text{IF}} = 2\pi/(8T_m)$ (i.e., with $\omega_{\text{IF}}/(2\pi) = 10$ MHz), this

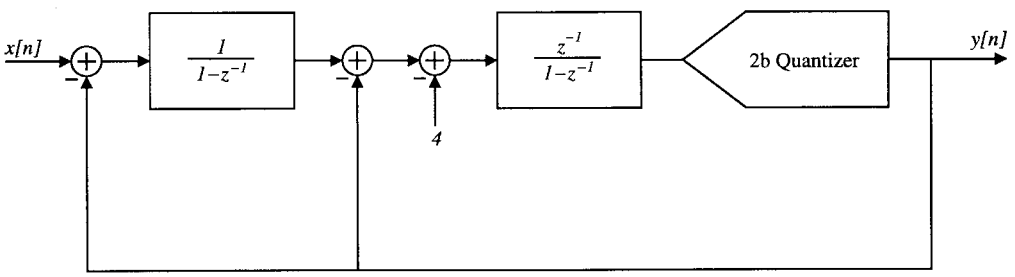


Fig. 8. The signal-processing equivalent of the $\Delta\Sigma$PLL redrawn to show it in the form of a second-order $\Delta\Sigma$ modulator.

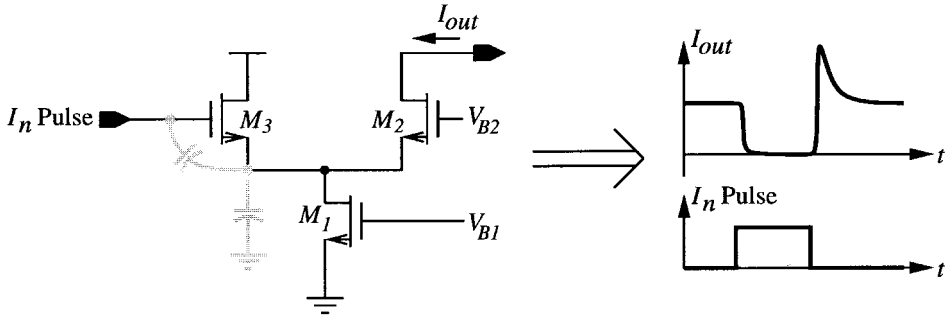


Fig. 9. The simple switched current source and the overshoot problem.

becomes

$$\frac{2\pi}{8T_m}(t_n - t_{n-1}) + \int_{t_{n-1}}^{t_n} \psi(\tau)\, d\tau = 2\pi. \tag{6}$$

With the definition

$$\hat{\psi}[n] = \int_{t_{n-1}}^{t_n} \psi(\tau)\, d\tau \tag{7}$$

it follows from (4) and (6) that $x[n] = (4/\pi)\hat{\psi}[n]$. Note that $\hat{\psi}[n]$ is the change in phase of $x_r(t)$ relative to $\omega_{\text{IF}}t$ during the nth sampling period δ_n. Thus, $\hat{\psi}[n]$ is the discrete-time derivative of the phase of $x_r(t)$ relative to $\omega_{\text{IF}}t$, which corresponds to nonuniformly sampled instantaneous frequency.

At first glance, it might seem that having $\hat{\psi}[n] = \psi(t_n)$ would be more desirable than the achieved relationship of (7). However, as demonstrated in Section IV, the slightly peculiar relationship between $\hat{\psi}[n]$ and $\psi(t)$ achieved by the $\Delta\Sigma$PLL gives rise to an extremely simple technique for correcting the distortion caused by nonuniform sampling.

C. Circuit Issues

The $\Delta\Sigma$FDC is robust with respect to nonideal circuit behavior because all of the analog components are positioned such that error they introduce is subjected to at least first-order noise shaping. This is evident from the analysis leading up to Fig. 8; the only components in Fig. 8 that arise from analog circuits (including the hard limiter) are the -4 offset, the $1/(1-z^{-1})$ integrator, and the quantizer. It is straightforward to verify that the transfer functions from each of these components to the output $y[n]$ all have at least one zero at dc and therefore provide at least first-order noise shaping. The remaining components in the figure are of digital origin, so they are not subject to analog circuit errors.

Only the positive current source is modulated by the input signal; the negative current source is turned on for a fixed duration each sample period δ_n. Therefore, relative deviations between the positive and negative currents do not affect the quantization noise shaping performance of the $\Delta\Sigma$PLL. However, as shown in [10], such errors do reduce the usable frequency range of the $\Delta\Sigma$PLL, so extreme mismatches between I_p and I_n should be avoided.

Deviations of I_p, C, and Δ from their ideal values result in a gain error in the signal processing associated with the charge pump. In the system of Fig. 8, this corresponds to a gain of $1+\alpha$ prior to the quantizer, where $\alpha = (I_p/f_m C\Delta) - 1$ is the gain error. Thus, the gain error has the same effect in the $\Delta\Sigma$PLL as it does in a second-order $\Delta\Sigma$ modulator; it causes the zeros and poles of the quantization noise shaping transfer function to deviate from their ideal positions. In the prototype design, no effort was made to have variations in I_p, f_m, C, and Δ track each other so as to minimize α because the expected variations gave rise to acceptable performance degradation. However, if such tracking were desired, it could be achieved by generating the ADC reference voltages using switched current sources to charge reference capacitors prior to each ADC sample.

Simulations and analysis indicate that the switched current sources must settle within approximately 6ns to ensure that the spurious-free dynamic range (SFDR) is greater than 85 dB. One approach to achieving fast settling is to use the simple switched current source shown in Fig. 9. In this circuit, when "I_n Pulse" is low, M_3 is off and M_1 and M_2 act as a cascode current source. When "I_n Pulse" goes high, the voltage at the drain of M_1 increases until M_2 is turned off. The circuit achieves very fast settling but is subject to turn-on overshoot, as depicted in Fig. 9. The reason for the turn-on overshoot is that as "I_n Pulse" goes low, the voltage at the

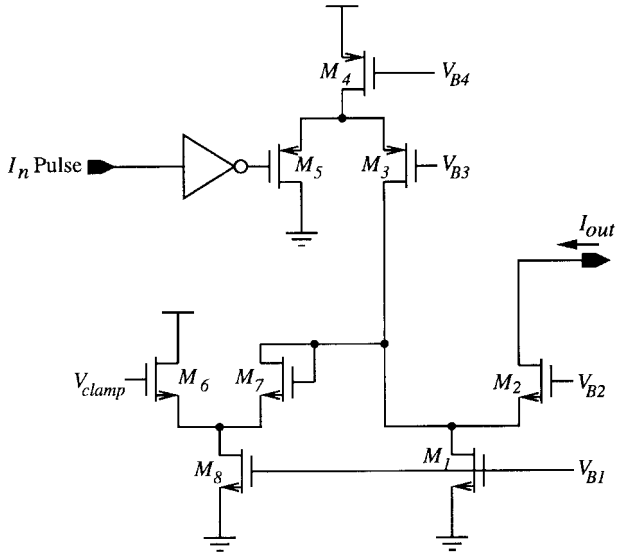


Fig. 10. A simplified schematic diagram of the negative switched current source used in the $\Delta\Sigma$PLL.

drain of M_1 temporarily goes too low because of the charge extracted through the parasitic gate–source capacitance of M_3. The overshoot corresponds to the additional current required to charge the parasitic capacitor at the drain of M_1 up to its steady-state voltage level.

The currents from the switched current sources get integrated by the summing capacitor in the $\Delta\Sigma$PLL, so overshoot has the same effect as using incorrect current values. By adjusting the relative sizes of the transistors in Fig. 9 so as to limit the significance of the charge injection through the gate–source capacitance of M_3, the overshoot problem can be avoided. However, simulations indicated that acceptably reducing the overshoot problem across process and temperature unacceptably increased the settling time. Therefore, the simple current source of Fig. 9 is not appropriate for the $\Delta\Sigma$PLL. Nevertheless, it does provide the basis for the improved switched current sources used in the prototype $\Delta\Sigma$PLL.

A simplified version of the negative switched current source used in the prototype $\Delta\Sigma$PLL is shown in Fig. 10. The circuit exploits the fact that the simple current source of Fig. 9 does not suffer from *turnoff* overshoot. Indeed, when the simple current source is turned off, the parasitic gate–source capacitance of M_3 actually aids the process because the charge injected through it increases the speed at which the voltage at the drain of M_3 rises. This increases the speed with which M_2 is turned off. The circuit of Fig. 10 makes use of this property by using the equivalent of two of the simple switched current sources of Fig. 9. The configuration is such that one of them is always turned off when the other is turned on so as to avoid overshoot. Specifically, when "I_n Pulse" is high, M_3 and M_7 are off, so M_1 and M_2 act as a cascode current source of approximately 18 μA. When "I_n Pulse" goes low, M_3 and M_4 act as a cascode current source of 20 μA, which causes the voltage at the drain of M_1 to rise to the point where M_2 is turned off. Transistors M_6–M_8 act as a clamping circuit that limits the voltage swing at the drain of M_1 to facilitate high-speed switching.

IV. The Nonuniform-to-Uniform Decimation Filter

The NTU decimation technique takes advantage of the property that the output sequence from the $\Delta\Sigma$PLL can be used to estimate both t_n and $\hat{\psi}[n]$. The NTU decimation filter estimates the actual sample times t_n from the $\Delta\Sigma$PLL output $y[n]$. For each *desired sample time* T_k, it performs two-point interpolation between the pair of adjacent actual sample times that straddle T_k to estimate the instantaneous frequency at T_k.

As shown in Fig. 11, the NTU decimation filter consists of three primary components: a conventional digital comb filter, a *nonuniform-to-uniform downsampler*, and a conventional 50-fold decimation filter. The details of these three components are explained in the remainder of this section.

A. Comb Filter

The $\Delta\Sigma$PLL output sequence $y[n]$ is filtered by a conventional digital filter with a so-called comb^2 transfer function given by

$$H(z) = \left(\frac{1 - z^{-4}}{1 - z^{-1}}\right)^2. \tag{8}$$

The purpose of the filter 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. The particular choice of $H(z)$ represents a good compromise with respect to these tradeoffs.[1] 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. The maximum passband magnitude of the comb filter is given by $H(e^{j\omega})|_{\omega=0} = 16$. Neglecting the passband droop of the comb filter, it follows from (2) that

$$y_f[n] = \frac{64}{\pi}\hat{\psi}[n] + e_f[n]$$

where $e_f[n] = h[n] * e[n]$ is the residual quantization noise after the filtering process. Alternatively, it follows from (4) that

$$y_f[n] = 128\left(1 - \frac{\delta_n}{8T_m}\right) + e_f[n]. \tag{9}$$

[1] As shown in [14], a comb filter of the form

$$H(z) = \left(\frac{1 - z^{-50}}{1 - z^{-1}}\right)^3$$

would be a better choice if the goal were simply to remove as much quantization noise as possible. In this case, the first stage of the decimation filter could downsample by a factor of 50 without significantly aliasing out-of-band quantization noise into the signal band. However, in the case of the $\Delta\Sigma$PLL, this filter would unacceptably distort the components of the oversampled signal that underwent spectral spreading. Instead, the looser filter given by (8) is used in conjunction with much less downsampling in the first stage.

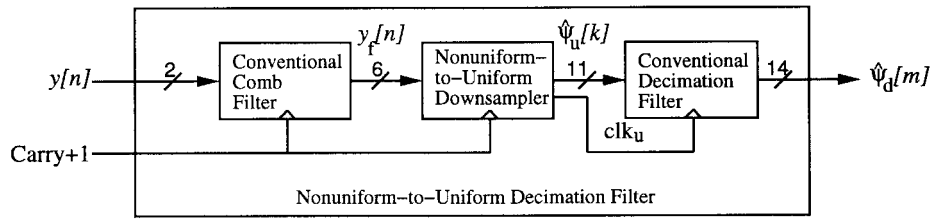


Fig. 11. The NTU decimation filter high-level structure.

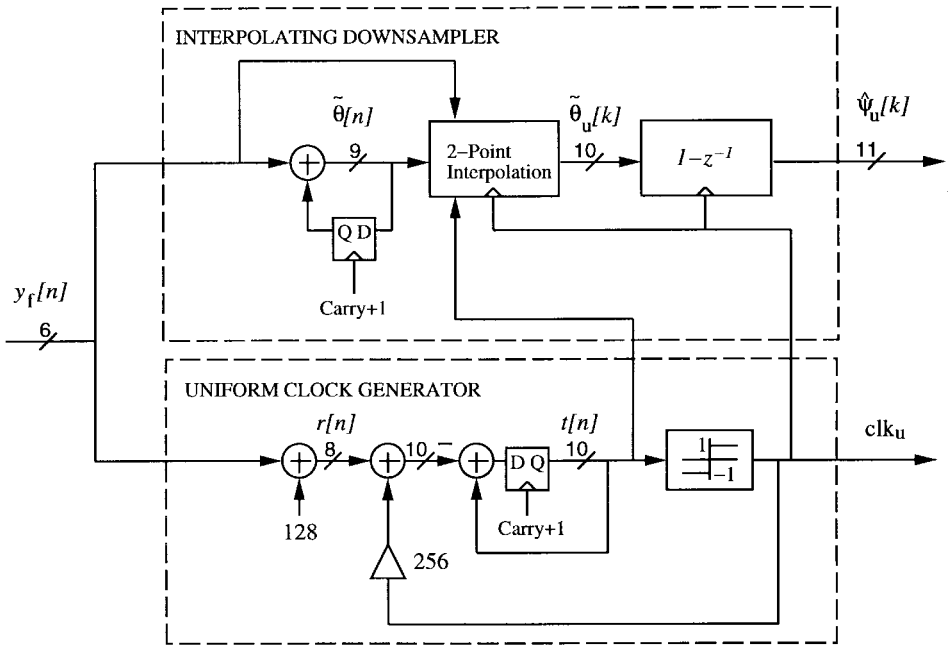


Fig. 12. The NTU downsampler.

B. NTU Downsampler

The NTU downsampler is shown at the register-transfer level in Fig. 12. It consists of two subblocks: the *uniform clock generator* and the *interpolating downsampler*. 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 $T_s = 32T_m$. This implies that $P = 4$ is the average downsampling ratio because $f_m = 1/T_m = 80$ MHz, and the average output sample rate of the $\Delta\Sigma$PLL is 10 MHz.

From Fig. 12 and (9), it follows that

$$r[n] = -128\left(\frac{\delta_n}{8T_m}\right) + 256 + e_f[n]. \tag{10}$$

The purpose of the register $t[n]$ shown in Fig. 12 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 Fig. 12 that the scaled output of the hard limiter (clk_u) is -256, which cancels the $+256$ dc offset of $r[n]$ from (10) 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 (10) and Fig. 12, the relation between the actual time estimate and the desired sample time is

$$t[n] = \frac{128}{8T_m}(t_n - T_k) + e_t[n] \tag{11}$$

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

$$\tilde{\theta}[n] = \frac{\pi}{64}\sum_{k=1}^{n} y_f[k] + \tilde{\theta}[0] \tag{12}$$

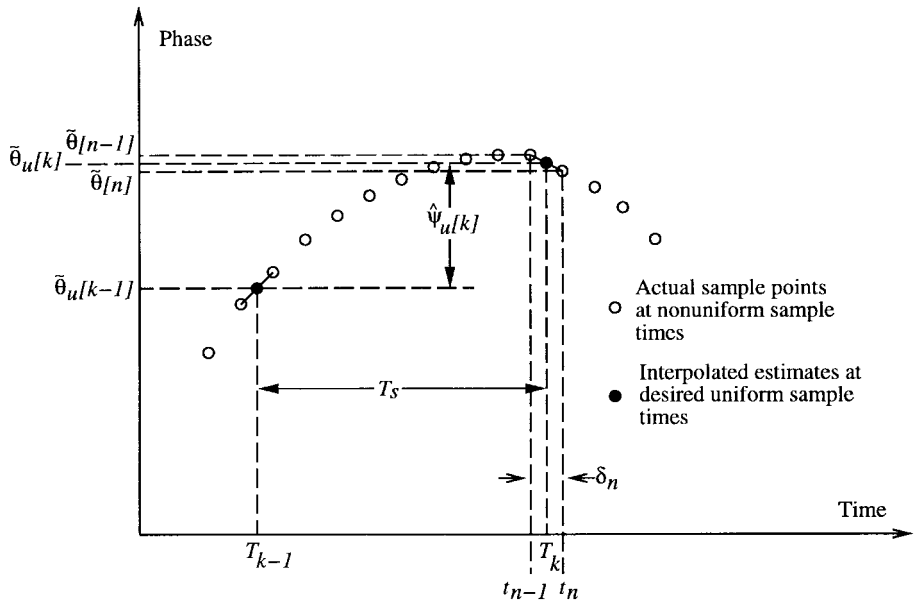


Fig. 13. The interpolation between actual sample points to estimate the desired downsampled values.

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.

The two-point interpolation expression is

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

which is depicted graphically in Fig. 13. From (9), (11), (12), and Fig. 12, it follows that

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

where the LSB normalization in (12) and the effects of quantization noise have been neglected. The system of Fig. 12 implements this form of the two-point interpolation expression.

Thus, only 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 2.5 MHz each. Moreover, the numbers of bits used for $y_f[n]$ and $t[n]$ are low (six and ten, respectively). Consequently, the computational requirements of the interpolation are not excessive.

As seen in Fig. 12, the $1 - z^{-1}$ operation performed upon $\tilde{\theta}_u[k]$ produces *uniform frequency estimates*, $\hat{\psi}_u[k]$. Ignoring quantization noise and interpolation error, it follows that $\hat{\psi}_u[k] = \tilde{\theta}_u[k] - \tilde{\theta}_u[k-1]$, which is proportional to the phase difference between times T_{k-1} and T_k. Thus, the sequence $\hat{\psi}_u[k]$ corresponds to frequency estimates at uniformly spaced sampling intervals.

C. Conventional Decimation Filter

The final component of the NTU decimation filter is a conventional 50-fold low-pass decimation and equalization filter. It removes the remaining out-of-band quantization noise and reduces the output sequence to the final desired sampling rate of 50 kSample/s. 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.

V. Experimental Results

As depicted in Fig. 14, the prototype $\Delta\Sigma$PLL was tested with and without the NTU decimation filter. Measurements without the NTU decimation filter were used to verify the expected structure of the quantization noise, and measurements with the NTU decimation filter were used to quantify the F/D conversion performance. As indicated in the figure, a software implementation of the NTU decimation filter was used. The bus widths and other processing details were implemented exactly as shown in Figs. 11 and 12, so identical results would be expected from an IC implementation.

The left PSD plot in Fig. 14 corresponds to the 2-b output of the $\Delta\Sigma$PLL driven by a 9 MHz sinusoid (i.e., a dc FM message signal). As expected, the plot is identical to that of a second-order $\Delta\Sigma$ modulator driven by a dc signal; the PSD is well approximated by (3). The right PSD plot in Fig. 14 corresponds to the output of the full $\Delta\Sigma$FDC driven by an FM signal with a 12 kHz sinusoidal message signal and a 500 kHz frequency deviation, i.e., the instantaneous frequency was given by $\psi(t) = 2\pi\Delta F \sin(2\pi ft)$ with $\Delta F = 500$ kHz and $f = 12$ kHz. For this case, the measured signal-to-noise-and-distortion ratio (SINAD) was 86.4 dB, and the measured SFDR was 90.5 dB.

In an ADC, the peak SINAD and SFDR are determined by increasing the amplitude of the input signal until the

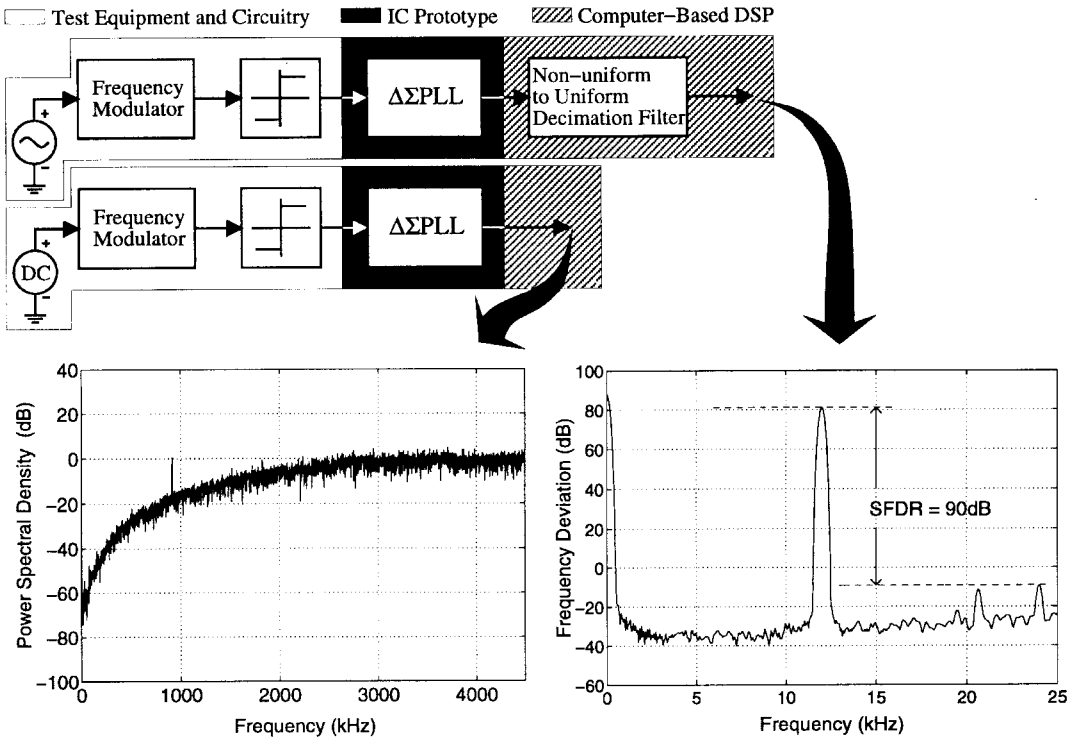


Fig. 14. Measurement schematic and PSD plots of representative measured data. The PSD of the $\Delta\Sigma$PLL output is in units of decibels/hertz relative to the $\Delta\Sigma$PLL quantization step size Δ^2 (for ease of comparison to the corresponding data from a conventional $\Delta\Sigma$) modulator. The PSD of the $\Delta\Sigma$FDC output is in units of decibels/hertz relative to 1 Hz^2 (because the output of the $\Delta\Sigma$FDC is a frequency estimate in units of hertz).

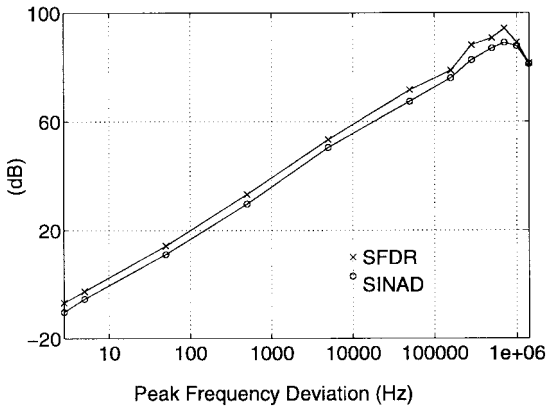


Fig. 15. Measured SINAD and SFDR versus frequency deviation (i.e., message signal amplitude).

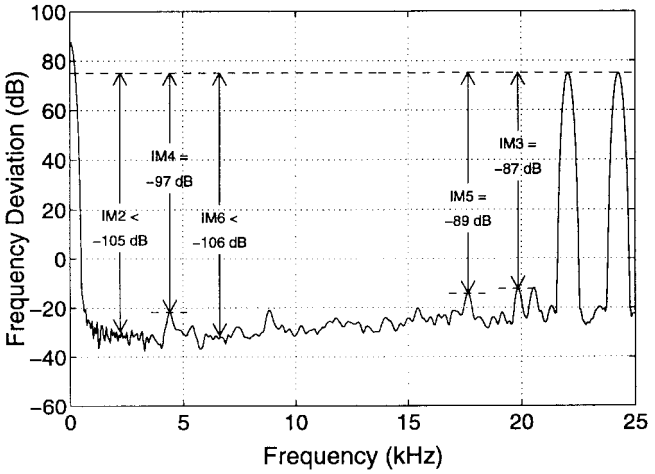


Fig. 16. Measured PSD for a two-tone message signal in units of decibels/hertz relative to 1 Hz^2.

measured SINAD and SFDR reach their maximum values. In an FDC, the same measurements are performed except the frequency deviation of the input signal is increased instead of the amplitude. Fig. 15 shows a plot of measured SINAD and SFDR values as functions of increasing frequency deviation. In each case, the input signal was an FM signal with a 12 kHz sinusoidal message signal. The resulting peak SINAD and SFDR were 89 and 94 dB, respectively. In general, the peak SINAD and SFDR were found to depend somewhat on the choice of the frequency and offset of the message signal used to generate the FM input signal. This is not surprising, as the output sample times of the $\Delta\Sigma$PLL are related to these quantities, so spurs that are out of band for some input signals end up in band for others. As part of the testing process, the worst case input signals were actively sought, and the resulting worst case peak SINAD and SFDR were found to be 85 and 88 dB, respectively. On the basis of measurements performed on numerous different input signals and different copies of the chip, a conservative estimate of typical values of peak SINAD and SFDR is 86 and 91 dB, respectively.

Additional measurements performed with two-tone message signals were consistent with these SINAD and SFDR values. For example, Fig. 16 shows the measured PSD of the $\Delta\Sigma$FDC output for an FM signal with instantaneous frequency given by $\psi(t) = 2\pi\Delta F_1 \sin(2\pi f_1 t) + 2\pi\Delta F_2 \sin(2\pi f_2 t)$, where $\Delta F_1 = \Delta F_2 = 250$ kHz, $f_1 = 22.1$ kHz, and $f_2 = 24.3$ kHz.

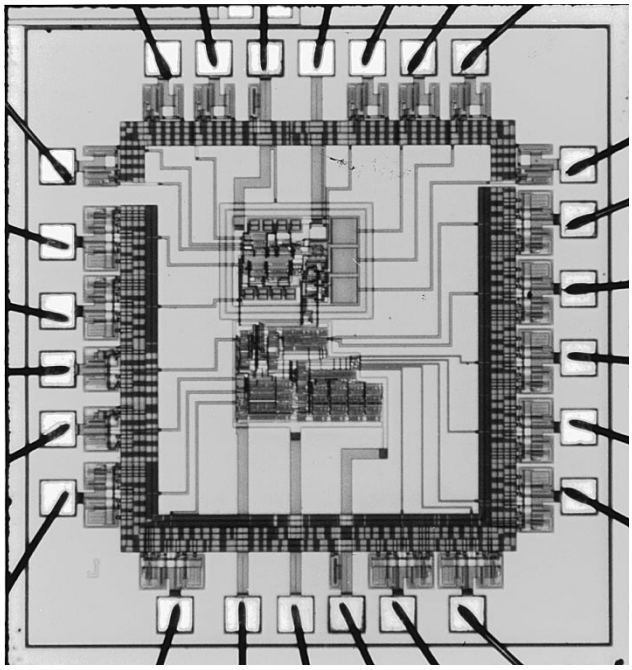


Fig. 17. Die photograph.

TABLE I
MEASUREMENT SUMMARY

Technology:	0.6μm single-poly CMOS
Input No-Overload Range:	8.9MHz to 10.0MHz
FDC Output Sample-Rate:	50kSample/s
Worst-Case FDC SINAD:	85dB
Worst-Case FDC SFDR:	88dB
Typical FDC SINAD:	86dB
Typical FDC SFDR:	91dB
External Clock Frequency:	80MHz
Supply Voltage:	5V
Power Dissipation (analog):	3mW
Power Dissipation (digital):	32mW
Total Power Dissipation:	35mW
Die Size:	4mm^2
Active Area:	0.6mm^2

As shown in the figure, the maximum in-band intermodulation product for this test was -87 dB.

For the prototype $\Delta\Sigma$PLL with the external clock frequency set to 80 MHz, the no-overload range is 8.9–10.0 MHz. Thus, provided the absolute instantaneous frequency of the $\Delta\Sigma$PLL input signal stays within this range, the ADC within the $\Delta\Sigma$PLL will not overload. As in a $\Delta\Sigma$ modulator, when the input signal exceeds the no-overload range, the data-conversion performance quickly degrades. The $\Delta\Sigma$PLL can be designed for different no-overload ranges by varying the external clock frequency, the number of counter bits, and the number of ADC bits, as described in [10].

As in a $\Delta\Sigma$ modulator, the no-overload range of the $\Delta\Sigma$PLL affects the signal-to-quantization noise ratio (SQNR) of the data-conversion process because it sets the maximum power of the signal component in the data-converter output. For example, consider the case of broadcast FM wherein the peak frequency deviation of the carrier is ± 75 kHz. In this case, only 14% of the full no-overload range of the prototype $\Delta\Sigma$PLL would be used, so approximately 17 dB of the peak SQNR would be wasted unless the remaining no-overload range were required to accommodate undesired frequency offsets.

As described above, the quantization-noise component in the data-converter output is determined by the ratio of the average $\Delta\Sigma$PLL output rate to the bandwidth of the message signal, i.e., the instantaneous frequency of the input signal, which in the case of the prototype is approximately 200. Hence, in contrast to $\Delta\Sigma$ modulators, the SQNR is decoupled from the bandwidth of the input signal and instead depends upon the bandwidth of the message signal.

A die photograph of the prototype $\Delta\Sigma$PLL is shown in Fig. 17. Two blocks of components are visible in the photograph. The top block, surrounded by guard rings, contains the analog components, and the bottom block contains the digital components. The details of the prototype are summarized in Table I.

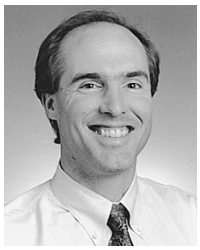
VI. CONCLUSION

A prototype $\Delta\Sigma$PLL based on the theory presented in [10] has been fabricated in a 0.6 μm, single-poly, CMOS process, and an efficient digital nonuniform-to-uniform decimation filter architecture has been proposed. Together, these components were shown to achieve 50 kSample/s frequency-to-digital conversion of a 10 MHz FM signal with a worst case SINAD of 85 dB and a worst case SFDR of 88 dB. The system performs limiter-discriminator FM demodulation and digitization. Relative to other approaches involving separate A/D conversion and FM discrimination, it offers the advantage of excellent performance with very low analog circuit complexity and only moderate digital complexity.

REFERENCES

[1] S. R. Norsworthy, R. Schreier, and G. C. Temes, *Delta–Sigma Data Converters: Theory, Design, and Simulation*. Piscataway, NJ: IEEE Press, 1997.
[2] A. K. Ong and B. A. Wooley, "A two-path bandpass $\Sigma\Delta$ modulator for digital IF extraction at 20 MHz," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 1997, pp. 212–213.
[3] M. Song, J. Park, W. Joe, M. J. Choe, and B. S. Song, "A fully-interated 5 MHz-IF FM demodulator," in *Proc. IEEE Custom Integrated Circuits Conf.*, May 1997.
[4] S. Jantzi, K. Martin, and A. Sedra, "A quadrature bandpass $\Sigma\Delta$ modulator for digital radio," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 1997, pp. 216–217.
[5] W. c. Lindsey and C. M. Chie, "A survey of digital phase-locked loops," *Proc. IEEE*, vol. 69, pp. 410–431, Apr. 1981.
[6] G. M. Bernstein, M. A. Lieberman, and A. J. Lichtenberg, "Nonlinear dynamics of a digital phase locked loop," *IEEE Trans. Commun.*, vol. 37, pp. 1062–1070, Oct. 1989.
[7] C. A. Pomalaza-Raez and C. D. McGillem, "Digital phase-locked loop behavior with clock and sampler quantization," *IEEE Trans. Commun.*, vol. COM-33, no. 8, pp. 753–759, Aug. 1985.
[8] B. C. Sarkar and S. Chattopadhyay, "Symmetric lock-range multi-level quantized digital phase locked FM demodulator," *IEEE Trans. Commun.*, vol. 38, no. 12, pp. 2114–2116, Dec. 1990.
[9] I. Galton and G. Zimmerman, "Combined RF phase extraction and digitization," in *Proc. IEEE Int. Symp. Circuits and Systems*, May 1993, pp. 1104–1107.
[10] I. Galton, "Analog-input digital phase-locked loops for precise frequency and phase demodulation," *IEEE Trans. Circuits Syst. Part II*, vol. 42, pp. 621–630, Oct. 1995.

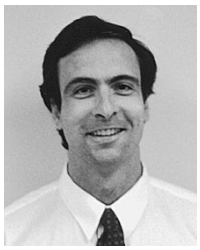
[11] R. D. Beards and M. A. Copeland, "An oversampled delta sigma frequency discriminator," *IEEE Trans. Circuits Syst.*, vol. 41, pp. 26–32, Jan. 1994.
[12] M. Hovin, A. Olsen, T. S. Lande, and C. Toumazou, "Delta-sigma modulators using frequency-modulated intermediate values," *IEEE J. Solid-State Circuits*, vol. 32, pp. 13–22, Jan. 1997.
[13] I. Galton, W. Huff, P. Carbone, and E. Siragusa, "A $\Delta\Sigma$PLL for 14b 50 kSample/s frequency-to-digital conversion of a 10 MHz FM signal," in *Dig. Tech. Papers 1998 IEEE Int. Solid-State Circuits Conf.*, vol. 41, pp. 366–367, Feb. 1998.
[14] J. C. Candy, "A use of double integration in sigma-delta modulation," *IEEE Trans. Commun.*, vol. COM-33, pp. 249–258, Mar. 1985.
[15] W. Huff and I. Galton, "Nonuniform to uniform decimation for delta-sigma frequency-to-digital conversion," in *Proc. 1998 IEEE Int. Symp. Circuits and Systems*, May 1998, pp. 365–368.
[16] U. L. Rohde, J. Whitaker, and T. T. N. Bucher, *Communications Receivers*, 2nd ed. New York: McGraw-Hill, 1997.



Ian Galton received the Sc.B. degree from Brown University, Providence, RI, in 1984 and the M.S. and Ph.D. degrees from the California Institute of Technology, Pasadena, in 1989 and 1992, respectively, all in electrical engineering.

He is currently an Associate Professor at the University of California (UC), San Diego. He was formerly with UC Irvine, Acuson, and Mead Data Central and has acted as a regular Consultant for several companies. His research interests involve integrated signal-processing circuits and systems for communications. He has received four patents.

Dr. Galton received the Caltech Charles Wilts doctoral thesis prize

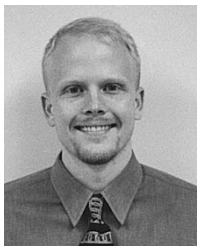


William Huff received the B.S. degree in electrical engineering from the University of California, Los Angeles, in 1990. He is currently pursuing the Ph.D. degree in electrical engineering at the University of California, San Diego.

From 1990 to 1996, he was an Analog Circuit Designer with Lear Astronics, Santa Monica, CA. His research interests include signal processing for communication systems and design of integrated circuits including delta–sigma frequency-to-digital converters and frequency synthesizers.

Paolo Carbone received the engineer's degree and the Dottorato di Ricerca degree from the University of Padova, Italy, in 1990 and 1994, respectively.

He joined the Electronics Engineering Department of the University "Roma Tre" in Rome, Italy, in 1994 as an Assistant Professor. Since 1997, he has been an Assistant Professor at the Istituto di Elettronica, University of Perugia, Italy. His research interests cover the application of digital signal-processing techniques to the measurement science and the issues related to their practical implementation.



Eric Siragusa received the B.S. degree *(summa cum laude)* in electrical engineering from the University of California, Irvine, in 1996. He is currently pursuing the Ph.D. degree in electrical engineering at the University of California, San Diego.

From 1996 to 1998, he was a Design Engineer with Newport Microsystems Corp., Irvine, CA, where he was involved in the system and circuit level design of mixed-signal communication IC's. His research interests include signal processing and IC design for communication systems and circuits.

Mr. Siragusa received a Center for Wireless Communication Graduate Student Fellowship in 1998.