A Continuous-Time Programmable Digital FIR Filter

Yee William Li, *Member, IEEE*, Kenneth L. Shepard, *Senior Member, IEEE*, and Yannis P. Tsividis, *Fellow, IEEE*

***Abstract*—In this paper, we describe the design and implementation of a continuous-time finite-impulse-response processor chain, which includes a 6-bit asynchronous ADC, an asynchronous digital core, and an 8-bit asynchronous DAC designed in TSMC 0.25-μm technology. The continuous-time, discrete-amplitude systems combine benefits associated with analog and digital systems. Discrete-amplitude representations leverage the noise immunity and robustness of digital implementations. Continuous-time, nonsampled operation prevents aliasing and reduces the in-band quantization noise associated with the aliasing of subharmonic components. We present measurement results demonstrating an audio low-pass filter with a bandwidth of 6.0 kHz.**

***Index Terms*—Asynchronous analog-to-digital conversion, continuous-time digital signal processing.**

I. Introduction

Conventional digital signal processing systems, which are discrete in time and discrete in amplitude (DTDA), are characterized by the deleterious effects of aliasing and quantization noise. Conventional analog systems, which process the signal continuously in time and amplitude (CTCA), do not suffer from these drawbacks. Analog systems, however, have high sensitivity to component tolerances and matchings, their dynamic range is comparatively low, and reconfigurability is limited and, usually, difficult. The merits and shortcomings of these systems are complementary, and this motivates a desire to find a signal processing system which combines the best attributes of both DTDA and CTCA systems. The focus of this study is to explore the relatively new area of systems which achieve exactly this combination of attributes by being discrete in amplitude but continuous in time (CTDA).

Fig. 1 shows a "four-quadrant" representation of signal processors which can be either discrete or continuous in amplitude or time [1]. Quantizers are used to convert signals from continuous to discrete amplitude, while samplers are used to convert signals from continuous to discrete time. The "first" and "third" quadrants represent the conventional analog (CTCA) and digital (DTDA) signal processing systems previously described. The "second" quadrant represents systems which are discrete-in-time but continuous-in-amplitude (DTCA). The best known example of DTCA systems are switched-capacitor (SC) circuits, which, like DTDA systems, have aliasing drawbacks. The "fourth" quadrant, representing systems which are discrete in amplitude but continuous in time (CTDA), have been largely unexplored and serve as the focus of this study. Such systems promise to eliminate aliasing and, as discussed in Section II, significantly reduce in-band quantization noise.

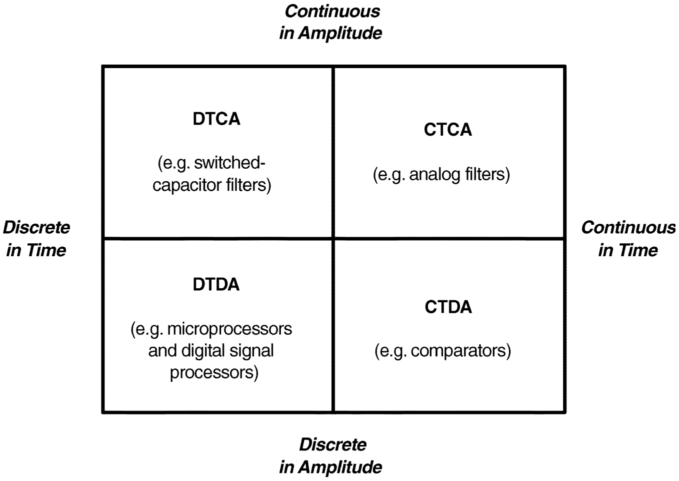


Fig. 1. Four signal processing domains: we can discretize in time and amplitude domains independently.

CTDA systems [2] are time-encoded; that is, they use the temporal positioning of samples to carry information [3], [4]. Such time-encoding is one representation of the sequence of action potentials in neurons [5] and has inspired theoretical studies of neuromorphic electronic systems [6]. Since CTDA systems are "clockless," asynchronous design techniques are a natural choice. However, it is important to note that asynchronous design techniques are usually employed in the design of DTDA system in which only the relative "ordering" of samples is preserved [7]; an implicit sample time is assumed based on the clock frequency of the analog-to-digital (ADC) converter. In contrast, this paper describes the design of an asynchronous digital processor that preserves the *time interval between samples*, since, in a CTDA representation, this carries information.

In this paper, we describe the detailed design of a complete CTDA finite-impulse-response (FIR) filter fabricated in a TSMC 0.25-μm CMOS technology. The system consists of a delta-modulated asynchronous ADC, an asynchronous digital FIR filter, and an asynchronous digital-to-analog converter (DAC) for analog waveform reconstruction as depicted in Fig. 2. We note that this system differs in many ways from that described in [8] and [9], most notably in the delta-modulated architecture and the use of true continuous-time digital processing. Such continuous-time digital processing was proposed

Manuscript received October 22, 2005; revised June 4, 2006. This work was supported in part by the National Science Foundation under Contract CCR-0086007.

Y. W. Li was with the Columbia Integrated Systems Laboratory, Department of Electrical Engineering, Columbia University, New York, NY 10027 USA. He is now with the Advanced Design Mixed-Signal Circuit Group, Intel Corporation, Hillsboro, OR 97124 USA (e-mail: ywli@ieee.org).

K. L. Shepard and Y. P. Tsividis are with the Columbia Integrated Systems Laboratory, Department of Electrical Engineering, Columbia University, New York, NY 10027 USA (e-mail: shepard@cisl.columbia.edu; tsividis@cisl.columbia.edu).

Digital Object Identifier 10.1109/JSSC.2006.883314

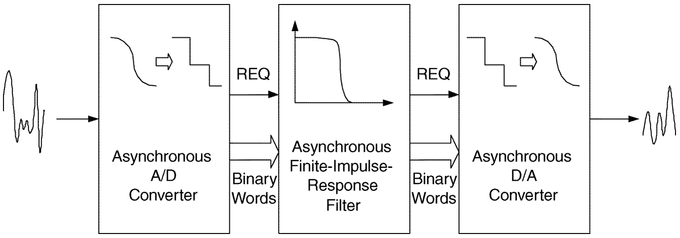

REQ
REQ
Asynchronous A/D Converter
Binary Words
Asynchronous Finite-Impulse-Response Filter
Binary Words
Asynchronous D/A Converter

Fig. 2. Continuous-time digital signal processor architecture.

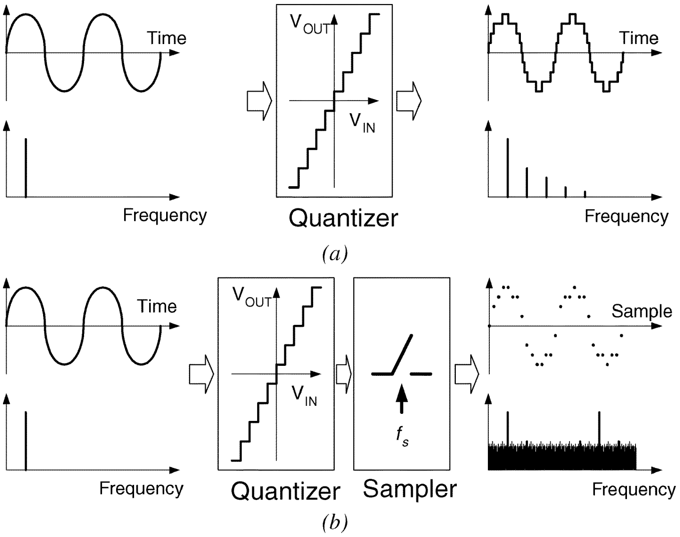


Fig. 3. Comparing between (a) CTDA and (b) DTDA systems.

in [2] and [10], but we consider the details of an integrated-circuit implementation here. In Section II, we discuss the potential benefits of CTDA systems in more detail. Section III present the overall chip architecture. The asynchronous ADC (A-ADC), digital core and asynchronous DAC (A-DAC) are introduced in Sections IV, V, and VI, respectively. The measurement results on the complete filter are presented in Section VII. Section VIII concludes the paper.

II. Benefits of Continuous-TIme Digital Signal Processing (CTDSP)

A. No Aliasing and Reduced In-Band Quantization Noise

When a signal is sampled at a sampling frequency f_s, the signal is converted into a discrete-time sequence. Aliasing results as there is no way to distinguish the frequency component $(\pm f_{\mathrm{signal}})$ from the aliased components $(p \cdot f_s \pm f_{\mathrm{signal}})$ where p is an integer. The elimination of sampling in CTDA systems results in the elimination of these aliasing effects.

The authors of [2] present a perspective to understand how quantization noise is reduced in a CTDA system, a discussion that we reproduce here briefly for completeness. If one inputs a sine wave (pure tone of frequency f) into an ideal quantizer, as shown in Fig. 3(a), higher order harmonics ($\pm mf$, where m is an integer) are introduced in the resulting output because of the nonlinearity of the quantizer. If the quantizer is followed by an ideal sampler with sample frequency f_s (which is equivalent to the more usual configuration of quantizer followed by sampler), then these higher order harmonics are aliased to frequencies $\pm pf_s \pm mf$, producing in-band quantization noise, as shown in Fig. 3(b). As a result, it is possible to think of quantization noise in conventional digital signal processing systems as resulting from the aliasing of the harmonic components resulting from quantization. Hence, if we eliminate sampling, no harmonics will be aliased to the in-band frequency spectrum, which is a significant advantage for continuous-time digital signal processing.

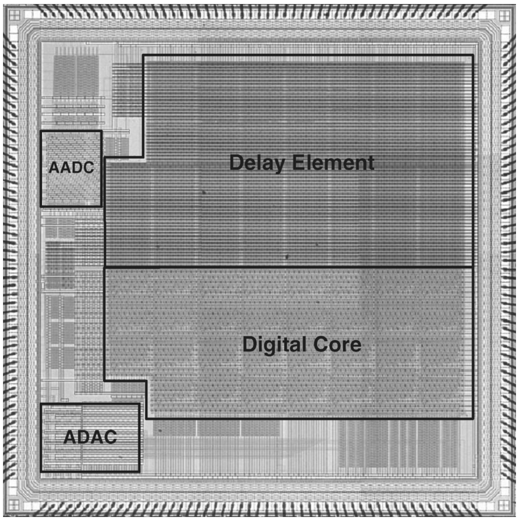


Fig. 4. Die photograph of the CTDA filter in a TSMC 0.25-μm process.

B. Potential Power Savings

In the conventional digital signal processor, we have to sample the signal at a rate at least twice the bandwidth of the signal (Nyquist rate) for reconstruction. As such, the power consumption is a function of the sample rate but independent of the actual spectral content of the input signal. Alternatively, if we can sample the signal only when there is activity, power savings may be realized in the subsequent signal processing. This was a benefit also recognized in the work of [8]. The asynchronous digital signal processor developed here has the property that the dynamic power dissipation is directly proportional to the bandwidth of the input signal.

III. Overall Chip Architecture

The chip is 25 mm^2, fabricated in the TSMC 0.25 μm logic process, and packaged in a 204-pin ceramic PGA; the die photograph is shown in Fig. 4 [11].

The design contains an ADC, a 16-tap digital FIR filter, and a DAC, all operating asynchronously. The ADC functions as a delta modulator [12], [13], outputting a single bit representing whether the new sample is one quantization level higher or lower than the preceding sample. Accumulation is done in the digital filter to 16 bits of precision. The result from the digital core is passed to an 8-bit current-steering DAC. In the chip, a bundled-data approach [14] is employed throughout, in which a request (**REQ**) signal accompanies the data. Unlike discrete-time asynchronous systems [15], [16], there is no acknowledgment handshake because no backpressure can be exerted to data movement. Because time intervals must be preserved, downstream

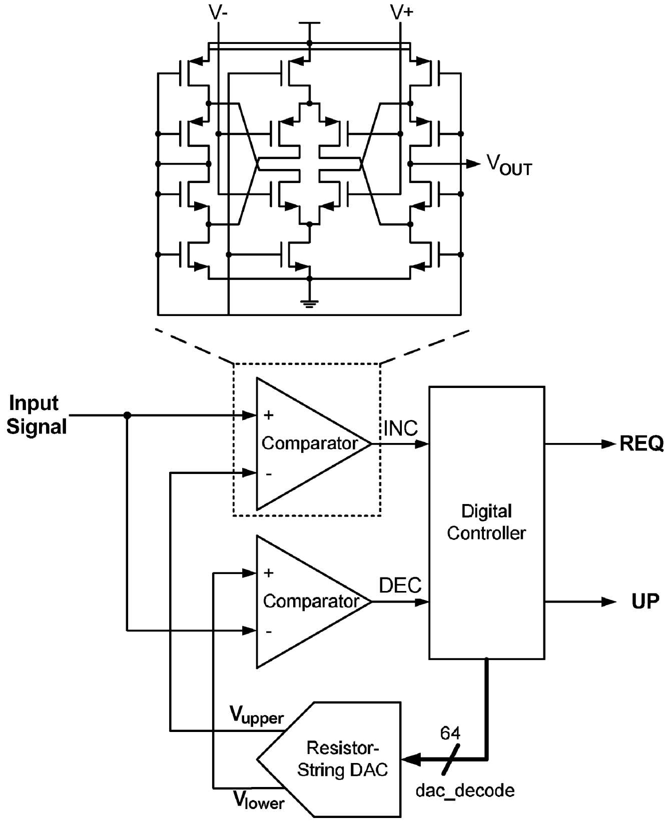


Fig. 5. A-ADC block diagram.

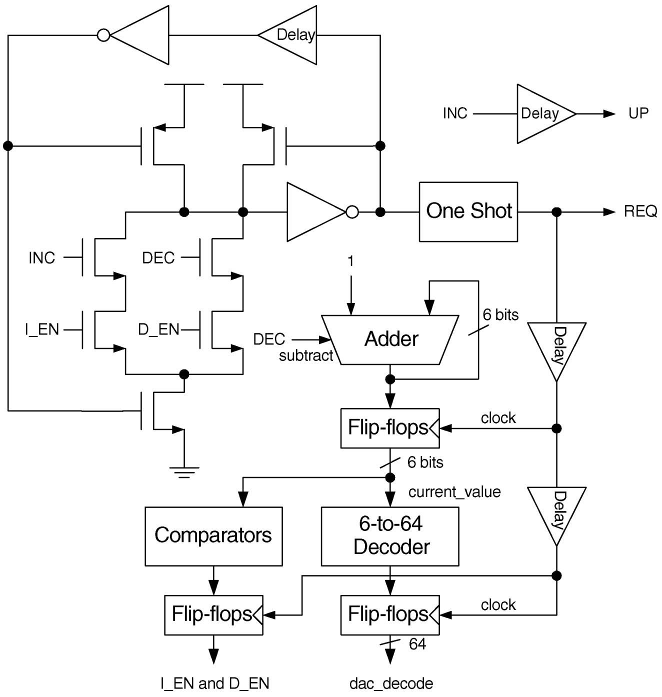


Fig. 6. Detailed implementation of the A-ADC controller.

processing elements must be able to immediately process data tokens.

IV. Asynchronous ADC (A-ADC)

Analog-to-digital conversion proceeds in a continuous-time manner in that samples are generated when the input analog signal crosses predetermined quantization levels [8], [17]. The delta-modulation ADC architecture implemented here (see Fig. 5) considerably simplifies the design of the digital FIR hardware by requiring only 1-bit signal representations for much of the datapath. A 6-bit resistor-string DAC generates two references voltages V_{upper} and V_{lower} separated by a least-significant-bit (LSB) resolution which "enclose" the instantaneous input voltage value (the one LSB resolution of the DAC is 16 mV). The DAC has a settling time of approximately 5 ns; this settle time is helped by the fact that codes are only changing by one LSB during loop operation. The comparator "front-ends" are implemented with the rail-to-rail comparator design shown in Fig. 5 [18], which gives a delay of between 7 and 15 ns across the full input common-mode voltage range from 1 to 2 V. Including the two levels of CMOS buffers that follow the front-end, the overall comparator gain is between 72 and 102 dB, providing for a comparator resolution of better than $V_{\text{resolve}} = 0.3$ mV. If the input voltage level crosses V_{upper}, **INC** goes to logic-1 (with **DEC** still zero). In response, the A-ADC controller (see Fig. 6) generates a **REQ** pulse with **UP** a logic-1. The n-bit **current_value** is incremented by one. Similarly, if the input voltage level crosses V_{lower}, **DEC** goes to logic-1 (with **INC** still zero). In response, the controller generates a **REQ** pulse with **UP** a logic-0, and **current_value** is decremented by one.

For slowly changing inputs ($f_{\text{input}} < (1/2\pi)(V_{\text{resolve}}/T_{\text{loop}})$, where T_{loop} is the loop delay of the ADC, that is, the combined delay of the controller, DAC, and comparator), it is possible for the comparator outputs (**INC** or **DEC**) to present mid-rail voltages to the controller. In this case, the dynamic pull-down stage at the input of the controller (see Fig. 6) may take longer to respond, resulting in variation in the delay to **REQ**. The only deleterious effect of this is some additional harmonic distortion in the A-ADC output.

The bandwidth of the input signal determines the maximum allowable T_{loop} of the ADC. Assume that a sine wave $A \cdot \sin(2\pi f_{\text{input}} t)$ with a peak derivative of $2\pi A f_{\text{input}}$ is input to the A-ADC with a full-scale input range A_{FS}. For n-bit data conversion, the minimum time T_{sample} for the input signal to traverse one LSB satisfies

$$T_{\text{sample}} = \frac{A_{\text{FS}}}{2^n 2\pi f_{\text{input}}}.$$

For correct operation, $T_{\text{loop}} < T_{\text{sample}}$, which indicates that the system is slew-rate limited. The above equation creates more stringent requirements on the data converter for higher resolutions (more closely spaced quantization levels). In the audio applications considered here, we assume that the input signal is bandlimited to 22 kHz. For $n = 6$ and $f_{\text{input}} = 22 \cdot 10^3$, $T_{\text{sample}} = 0.226$ μs. By contrast, for 8-bit operation, T_{sample} is only 28 ns.

In the A-ADC controller, the **I_EN** and **D_EN** signals are deasserted when the input signal exceeds full-scale to prevent new data from being sent from the ADC. The self-resetting initial stage has a reset delay (T_{reset}) that ensures that, once

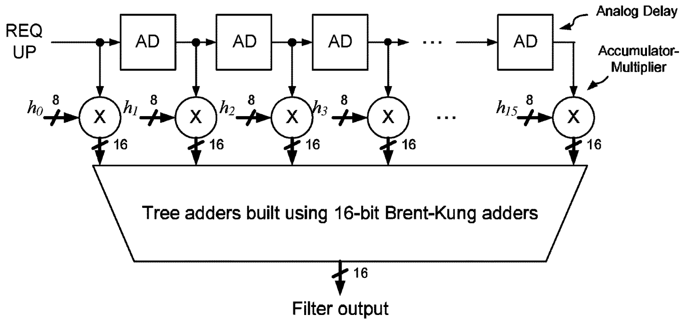


Fig. 7. Asynchronous FIR filter block diagram.

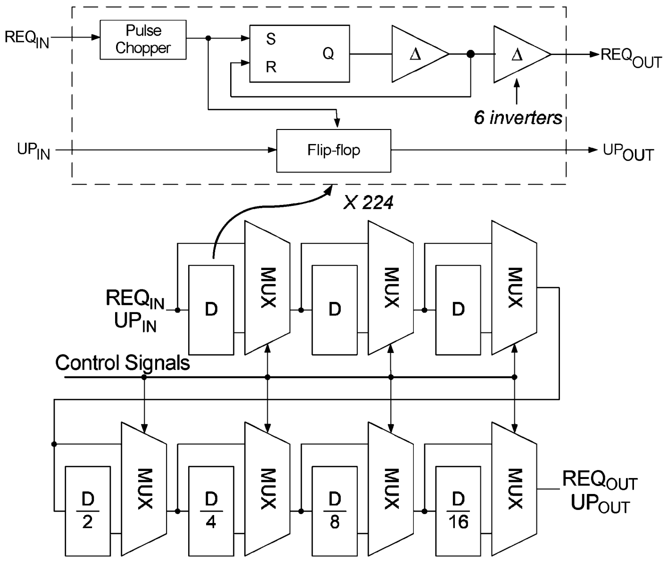


Fig. 8. Digitally tunable analog delay line.

this stage has evaluated (by the action of the **INC** or **DEC** signals), it cannot reevaluate again for a time T_{reset}. This ensures that spurious requests are not generated while the ADC is settling. In particular, to ensure this, $T_{\text{reset}} > T_{\text{loop}}$. In our design $T_{\text{reset}} \cong 30$ ns, while $T_{\text{loop}} \cong 20$ ns. The one-shot "chopper" circuit generates the **REQ** pulse, which is approximately 500 ps in width. This pulse is also used to trigger the internal state latches of the controller [19].

V. Asynchronous FIR Filter

The dataflow portion of the digital filter is shown in Fig. 7. The 1-bit **UP** signal from the A-ADC is accompanied by the 1-bit request signal (**REQ**). The strobe, which follows the data through the digital processing, allows for data-independent matched timing through the dataflow elements. Since the timing interval between data elements carries information, this interval must be preserved in computation. This datapath implements the FIR function

$$y(t) = \sum_{k=0}^{15} h_k x(t - kT_{\text{delay}})$$

by means of three main components: analog delay (**AD**) blocks, which provide for a continuous-time delay of the A-ADC output; multiplier-accumulators; and a final adder. T_{delay} is the delay of one of the **AD** blocks and the h_k are the 8-bit filter coefficients. The multiplier-accumulator block generates 16-bit results which are combined in the final adder.

Analog Delay Elements: The **AD** blocks control the delay of the **REQ** signals which "synchronize" the 1-bit output of the delta-modulator ADC through flip-flops. Design of the **AD** blocks presents considerable challenges. Given the low bandwidth of the input signals, this delay block must present a nominal delay time (T_{delay}) which is large compared to the fanout-of-four (FO4) delay of this technology (T_{FO4}). Furthermore, this **AD** block must have sufficient granularity, that is, have enough delay elements, to ensure that it has the capacity to store the requisite number of **REQ** events. In particular, for n-bit operation, there could be as many as

$$\frac{T_{\text{delay}}}{T_{\text{sample}}} = T_{\text{delay}} 2^{n-1} 2\pi f_{\text{input}} \qquad (1)$$

request pulses in the line, where f_{input} is the bandwidth of the band-limited input signal. For $T_{\text{delay}} = 12.5$ μs, $f_{\text{input}} =$ 22 kHz, and $n = 6$, 56 pulses could be in the delay line at once, while for 9-bit operation there could be as many as 446 request pulses in the line.

The detailed implementation of one of the **AD** blocks is shown in Fig. 8. In our design $D = 6.4$ μs, and, with multiplexer-based control, we can digitally set any delay between 0.4 μs and 25.2 μs on steps of 0.4 μs. The multiplexer control signals are set through a scan chain. These multiplexer controls also allow these delays to be calibrated in the presence of process variations; separate controls for each delay element allow tuning in the presence of intradie variability. A D unit of delay is implemented with 224 basic delay elements shown in the inset of Fig. 8, each providing approximately 30 ns of delay. It is intended that each of the basic delay blocks accommodate at most one request pulse. These elements, according to (1), can, therefore, handle 9-bit operation $(n = 9)$ for a delay of at least $2D$. The Δ delay blocks of Fig. 8 are implemented with six inverters, which, by choice of nonminimum length devices, each have a stage effort [20] of 100, providing a delay of approximately $25T_{\text{FO4}}$.[1] Because an inverter will inevitably have a small beta ratio skew, either by unintended design or process variations, a pulse when passed through a large number of inverters will, in general, be runted or could be filtered out entirely. As a result, regeneration is required, leading to the use of the SR-latch and one-shot in the basic delay block of Fig. 8. This preserves a pulse width of approximately Δ as each request pulse passes through a delay block.

Accumulator-Multiplier: The accumulator-multiplier function of Fig. 9(a) can be implemented without hardware multipliers as shown in Fig. 9(b). This results from observing that

$$\begin{aligned} y[n] &= (x[n-1] \pm 1) \cdot h_k \\ &= x[n-1] \cdot h_k \pm h_k \\ &= y[n-1] \pm h_k \end{aligned} \qquad (2)$$

[1] A more power-efficient approach to the high-stage effort buffers would be to use current-starved inverters [21]. These were, unfortunately, not pursued for this implementation.

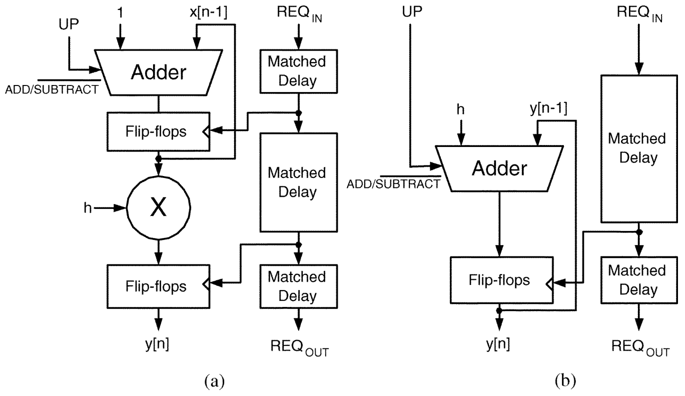


Fig. 9. (a) Accumulator-multiplier behavior can be implemented (b) with only an adder.

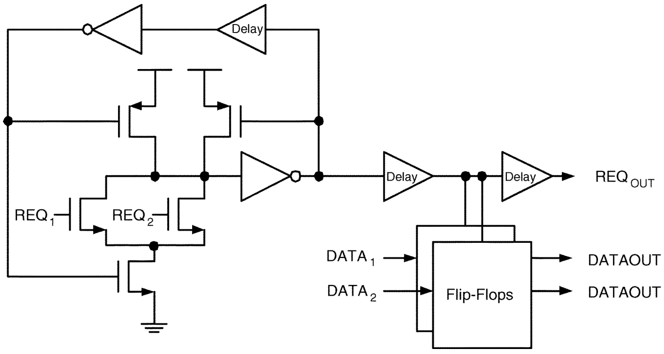


Fig. 10. Join controls used in the adder tree.

where the index n denotes the current sample and $n-1$ denotes the preceeding sample. Despite the use of "discrete-time" notation, these samples are, of course, nonuniformly spaced. Delay blocks are used to time the traversal of the **REQ** signal through the multiplier block. By using the **REQ** signal to "synchronize" the result, delay data dependencies are removed. This also ensures that the delays through each of the accumulator-multiplier blocks is matched, which is important to correct operation.

Final Adder: A Brent–Kung adder tree is used to add the 16 partial sums of the accumulator-multiplier blocks. As in the accumulator-multiplier block, the **REQ** signal timed through a delay line is used to sequence the data through a set of latch-bounded pipeline stages. The pipeline stage delay (T_{pipe}) must be less than the shortest interval between any two samples (T_{sample}), that is, $T_{\mathrm{pipe}} < T_{\mathrm{sample}}$. If this is not the case, data "waves" may collide within a single pipeline stage. We note again that backpressure through an acknowledgment signal cannot be applied, since the time spacing between samples must be preserved.

Datapath joins in the Brent–Kung tree introduce unique challenges to the design. Consider the case of two samples which are separated by exactly T_{delay}, the delay of a single analog delay (**AD**) element. In this situation, the two requests will arrive at the inputs of the adder at exactly the same instance. By this example, it is clear that the inputs to any of the blocks in the Brent–Kung tree can arrive arbitrarily close together. In this case, a mechanism must exist to allow one of the requests to be processed and to discard the other. In particular, this will happen if the two requests are closer together than $T_{\mathrm{separation}}$, where $T_{\mathrm{separation}} > T_{\mathrm{pipe}}$. Fig. 10 shows the join "control" circuit used for this purpose. $T_{\mathrm{separation}}$ is defined by the self-resetting loop delay. In the case of "collisions" resolved by the join controls, it is possible for a "new" value of $DATA_1$ or $DATA_2$ to be missed until the next request can be processed, resulting in potential errors in the least-significant bits of the result. Infrequent metastability of these least significant bit positions in the flip-flops is also possible in the case of collisions.[2] In this application, these infrequent "dropped samples" would correspond to very closely samples presented to the DAC; as a result, the only result of this action is harmonic distortion that is well outside the signal bandwidth of interest and can be easily removed by a reconstruction (or smoothing) filter if not already removed by the finite bandwidth of the DAC. In our design, $T_{\mathrm{separation}} = 6$ ns and $T_{\mathrm{pipe}} = 1.5$ ns.

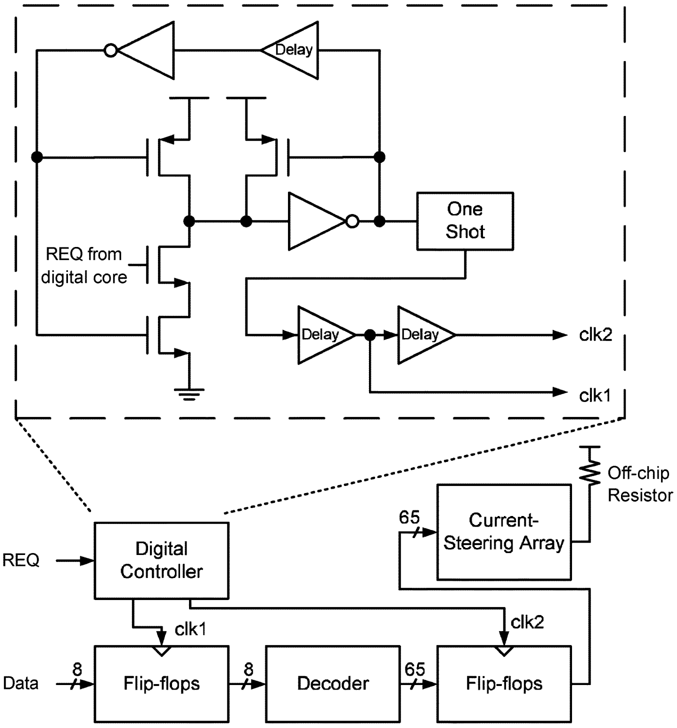


Fig. 11. A-DAC block diagram.

A programmable shifter (controlled with scan-chain bits) exists at each stage of the adder tree to normalize the result and prevent overflow.

VI. Asynchronous DAC (A-DAC)

The eight most significant bits of the asynchronous adder are presented to the DAC to produce the analog output. The A-DAC is implemented with a current-steering architecture as shown in Fig. 11. The digital controller (Fig. 11) is similar to the one in the ADC, employing a self-resetting initial stage that ensures that the DAC output has stabilized before a new request is processed. The worst case settling time of the DAC, which determines the minimum possible time between samples, is approx-

[2] The probability of a collision can be approximated by $T_{\mathrm{separate}}/T_{\mathrm{delay}}$ assuming a random arrival time for the two inputs to the join controls. For a typical delay time of 12.5 μs in our case, this corresponds to a probability of 5×10^{-4}.

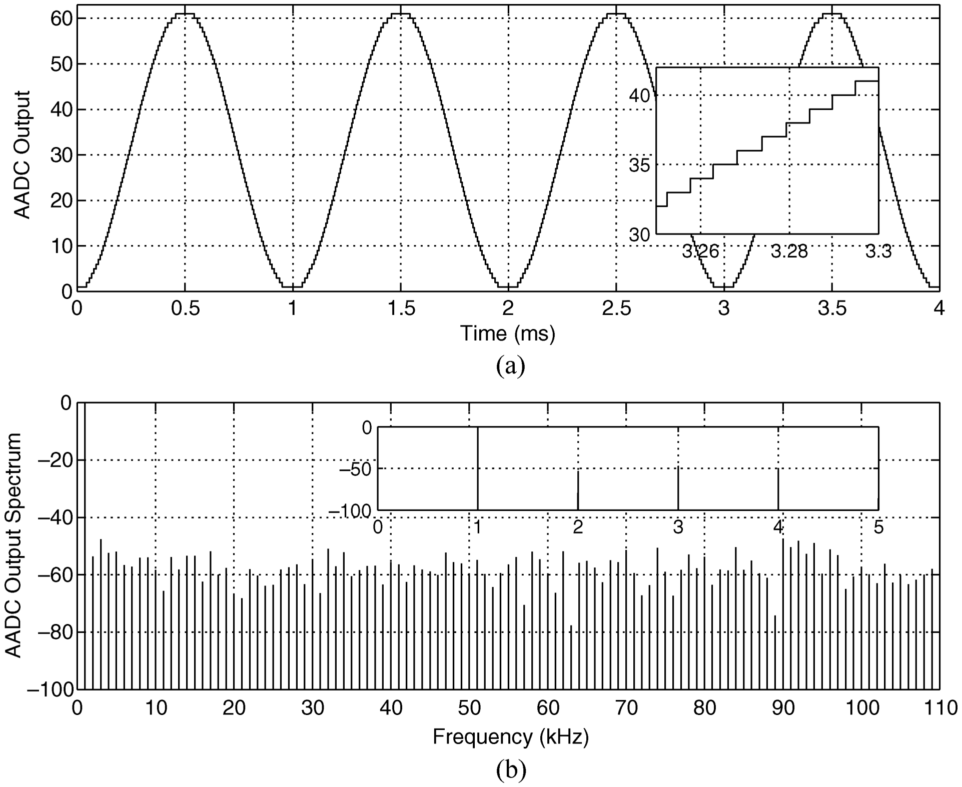


Fig. 12. (a) Measured output of the A-ADC for a 1-kHz sinusoidal input. (b) Corresponding frequency spectrum of the A-ADC output.

TABLE I
SUMMARY OF MEASURED SYSTEM CHARACTERISTICS

System	Area	25
	Power: 1 to 22 kHz (0 dBFS tone)	42.2 to 278 mW
	SNDR (over 55 kHz bandwidth)	35.0 to 52.3 dB
AADC	Area	0.27
	SNDR (over 55 kHz bandwidth)	38.6 to 53.7 dB
	Number of bits	6
	Power: 1 to 22 kHz / -0.1 dBFS tone	16.1 to 17.0 mW
ADAC	Area	0.42
	Number of bits	8
	Power: 1 to 22 kHz / -0.1 dBFS tone	25.5 to 30.8 mW
Delay elements	Area	6.60
	Power: 1 kHz / 0 dBFS tone	9.84 mW
Digital core	Area	5.06
	Power: 1 to 22 kHz (0 dBFS tone)	2.0 to 42.0 mW

imately 40 ns, corresponding to the digital code switching from all zeros to all ones. The decoder converts the two's complement output from the digital core into the decoder representation for the current-source array. Thermometer and binary codes are mixed in the DAC to balance complexity and matching issues. The two LSBs are binary coded while the six MSBs are thermometer coded [22].

VII. HARDWARE MEASUREMENT RESULTS

Measured properties of the CTDA filter are summarized in Table I.

A. A-ADC Performance

To characterize the performance of the ADC, the accumulated (integrated) delta-modulated output of the ADC is captured with the sample arrival time captured and quantized to a resolution of $\Delta t = 4$ ns. In order to assess the performance without the interference of the digital core to the ADC, this test is carried out with the filter and DAC turned off. Fig. 12(a) shows the binary output from the ADC with a 1-kHz sinusoidal input at 0 dBFS; the inset clearly shows the output is quantized. Fig. 12(b) is the corresponding frequency spectrum, computed from a fast Fourier transform (FFT) with a Welsh window function [23] to a frequency resolution of 25 Hz. The inset depicts the features of the spectrum in the range up to 5 kHz. Only the discrete harmonics of the input frequency are evident. By contrast, the simulated spectrum of a conventional, sampling "ideal" 6-bit ADC with a sampling rate of $5 \cdot 22$ kHz (2.5 times the Nyquist rate) is shown in Fig. 13. In this case, the aliased component of the input tone is evident at 109 kHz as is aliased harmonic distortion which appears as in-band quantization noise.

B. Asynchronous Digital Core and DAC Performance

The recorded **UP** and **REQ** signals captured in the above test with 4-ns time resolution is fed to the filter and subsequently to the DAC. The A-ADC is turned off for this test. With the filter programmed to be an all-pass, Fig. 14(a) shows the resulting filter output. The "glitches" in the waveform observed near 0.8, 1.8, 2.8, and 3.8 ms are due to collision in the join controls of the final adder, as described above. We verified this by reconfiguring the filter to have only a single tap feeding the final adder; in this case, the glitches are no longer present. The

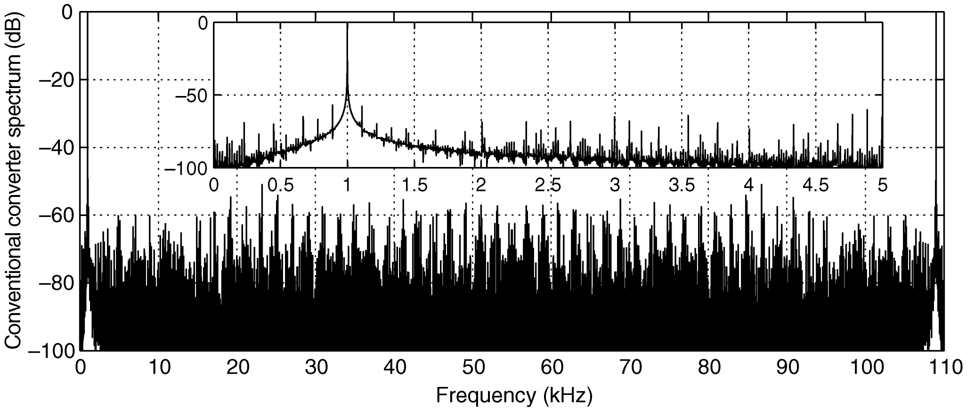


Fig. 13. Simulated frequency spectrum of a conventional, sampling ideal 6-bit converter operating at $f_s = 110$ kHz (inset: expanded view up to 5 kHz).

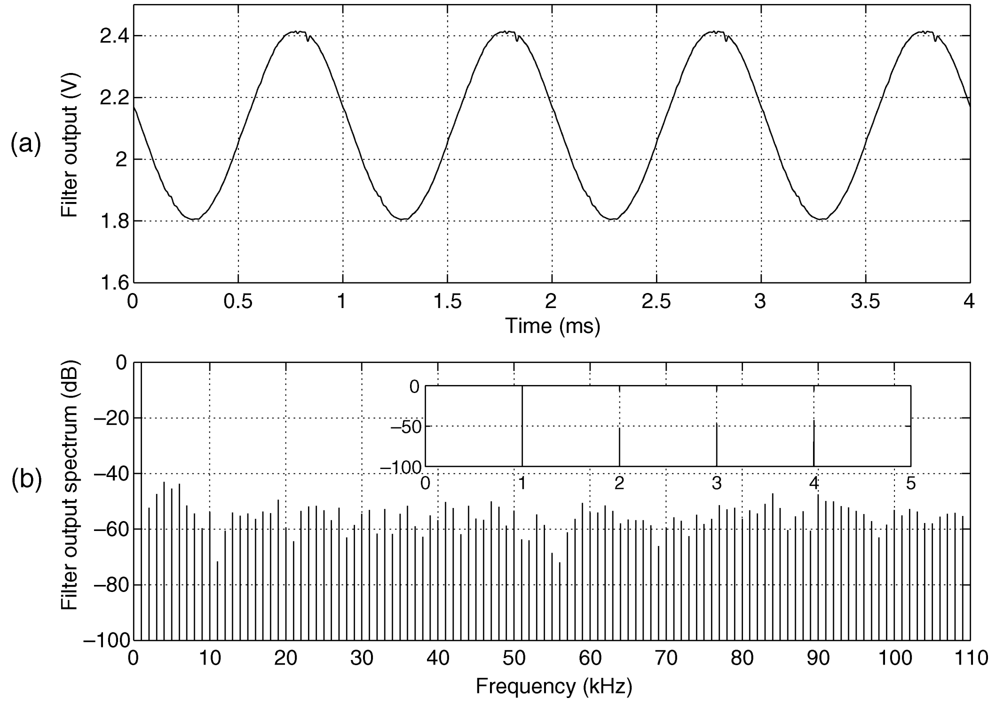


Fig. 14. (a) Measured output of the filter output with a 1-kHz sinusoidal input at the A-ADC. (b) Corresponding frequency spectrum of the filter output.

corresponding spectrum in Fig. 14(b) shows that these imperfections, while resulting in additional harmonic distortion in the output waveform, are mostly out-of-band. The "noise floor" of the spectrum is still about -140 dB.

Fig. 15 shows the measured signal-to-noise-plus-distortion ratio (SNDR) (for a 55-kHz band) at both the output of the A-ADC and the output of the A-DAC. SNDR is shown as a function of input signal frequency for a full-scale sinusoidal input signal. The "classical" quantization noise value, as given by $6.02n + 1.76$ dB for an n-bit converter, is shown for 6 bits (37.8 dB). The SNDR improves for input tones higher in the target bandwidth, since fewer in-band harmonics appear. The "steps" around 18 kHz occur as the third harmonic of the input frequency are pushed out of band. Similar steps around 13 kHz occur as the fourth harmonic is pushed out of band. These harmonics are clearly more pronounced at the A-DAC output, due to additional harmonic distortion introduced there. At 22 kHz, the SNDR of the A-ADC and A-DAC are 53.7 and 52.3 dB, respectively, which is better than the quantization noise floor for a conventional 8-bit converter (50 dB) even though only 6-bit quantization is employed here.

C. Overall Filter Characteristics

To characterize the entire filter, the filter is programmed to be low-pass with the cutoff at 6.0 kHz; the coefficients are obtained using the window method with a Hamming window [23]. Fig. 16 shows the magnitude of the filter transfer function as measured from 100 Hz to 25 kHz. At 14.5 kHz, the attenuation is more than -48 dB, reaching the resolution limit of the 8-bit A-DAC. The measured results are compared in Fig. 16 with the simulated ideal filter response.

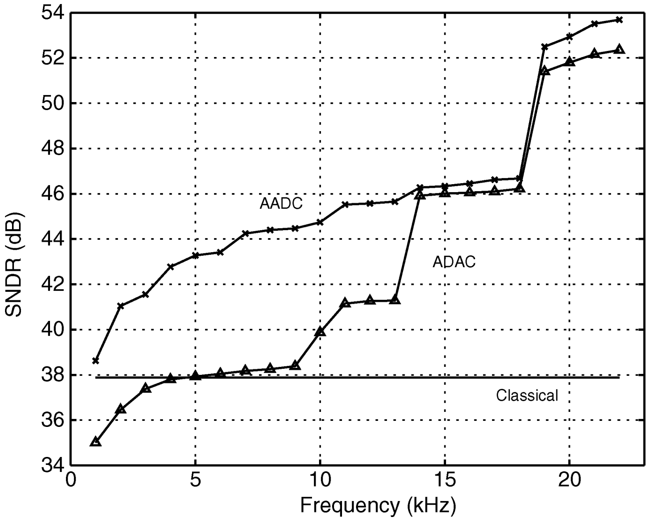


Fig. 15. Measured SNDR at the output of the A-ADC and A-DAC as a function of the frequency of a full-scale input tone.

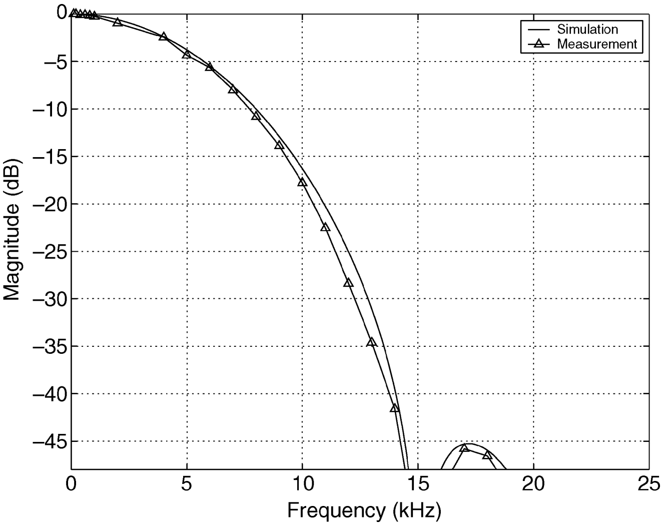


Fig. 16. Measured and simulated low-pass filter transfer characteristic.

D. Power Measurement

The power consumption of the CTDA filter implemented here is a strong function of the spectral content of the input signal. Fig. 17 shows the measured power consumed by various components of the filter as a function of the frequency of a full-scale input sinusoid. The delay elements and digital core show a close to linear dependence of power on input frequency. For the A-ADC and A-DAC, which are dominated by static bias currents, the power is virtually independent of frequency. The power is dominated by the delay elements, which are designed to support 9-bit resolution from the A-ADC. With identical design of the delay stages, If only 6-bit resolution were required, the delay of each basic delay element could be increased by eight with one-eighth the number of delay stage employed [see (1)]. By employing current starved inverters to achieve this delay increase, a factor-of-eight reduction in the dynamic power dissipation of the delay elements would be achievable. This scaled-back power dissipation is also noted in Fig. 17. With this scale-back, the area consumed by the delay elements would be reduced from 6.6 mm^2 to less than 1 mm^2.

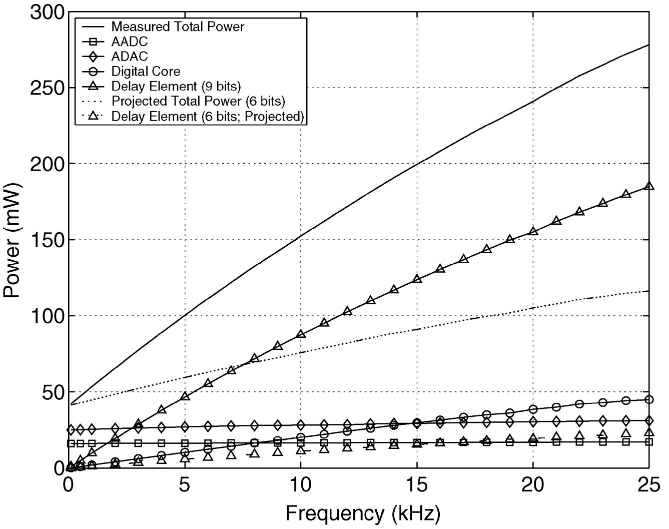


Fig. 17. Power measurements on components of filter a function of input frequency, and projected power for a true 6-bit implementation.

VIII. Conclusion

We have demonstrated the design of a continuous-time, discrete-amplitude digital signal processor. Such a design leverages the noise immunity and robustness of digital systems with nonsampled operation to prevent aliasing. An initial prototype design has demonstrated operation as a programmable audio filter. We have measured a significant reduction in in-band quantization noise (over 15 dB for 6-bit quantization) due to the elimination of aliasing. In addition, power dissipation shows a linear relationship with input signal bandwidth, thus demonstrating the potential for dynamic power reduction with this approach.

Acknowledgment

The authors would like to thank Prof. S. M. Nowick for several helpful discussions.

References

[1] R. Sarpeshkar, “Analog versus digital: Extrapolating from electronics to neurobiology,” *Neural Computation*, vol. 10, pp. 1601–1638, 1998.

[2] Y. Tsividis, “Digital signal processing in continuous time: A possibility for avoiding aliasing and reducing quantization error,” in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, May 2004, vol. II, pp. 589–592.

[3] H. Inoue, T. Aoki, and K. Watanabe, “Asynchronous delta modulation system,” *Electron. Lett.*, vol. 2, pp. 95–96, 1966.

[4] J. Foster and T.-K. Wang, “Speech coding using time code modulation,” in *Proc. IEEE SoutheastCon*, 1991, vol. 2, pp. 861–863.

[5] E. D. Adrian*, The Basis of Sensation: The Action of the Sense Organs*. London, U.K.: Christophers, 1928.

[6] A. A. Lazar and L. T. Toth, “Time encoding and perfect recovery of bandlimited signals,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Hong Kong, 2003, vol. VI, pp. 709–712.

[7] Y. W. Li, G. Patounakis, K. L. Shepard, and S. M. Nowick, “High-throughput asynchronous datapath with software-controlled voltage scaling,” *IEEE J. Solid-State Circuits*, vol. 39, no. 4, pp. 704–708, Apr. 2004.

[8] E. Allier, G. Sicard, L. Fesquet, and M. Renaudin, “A new class of asynchronous A/D converts based on time quantization,” in *Proc. Int. Symp. Asynchronous Circuits Syst.*, 2003, pp. 196–205.

[9] F. Aeschlimann, E. Allier, L. Fesquet, and M. Renaudin, “Asynchronous FIR filters: towards a new digital processing chain,” in *Proc. Int. Symp. Asynchronous Circuits Syst.*, 2004, pp. 198–206.

[10] Y. Tsividis, "Continuous-time digital signal processing," *Electron. Lett.*, vol. 39, no. 21, pp. 551–1552, Oct. 2003.

[11] Y. W. Li, K. L. Shepard, and Y. P. Tsividis, "A continuous-time programmable digital FIR filter," in *Proc. IEEE Custom Integrated Circuits Conf.*, Sep. 2005, pp. 695–698.

[12] G. Lockhart, "Digital encoding and filtering using delta modulation," *Radio and Electronic Engineer*, vol. 42, no. 12, pp. 547–551, Dec. 1972.

[13] N. Kouvaras, "Operations on delta-modulated signals and their application in the realization of digital filters," *Radio and Electronic Engineer*, vol. 48, pp. 431–438, 1978.

[14] S. Hauck, "Asynchronous design methodologies: An overview," *Proc. IEEE*, vol. 83, no. 1, pp. 69–93, Jan. 1995.

[15] S. Schuster, W. Reohr, P. Cook, D. Heidel, M. Immediato, and K. Jenkins, "Asynchronous interlocked pipelined CMOS circuits operating at 3.3–4.5 GHz," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2000, pp. 292–293.

[16] J. Tierno, A. Rylyakov, S. Rylov, M. Singh, P. Ampadu, S. Nowick, M. Immediato, and S. Gowda, "A 1.3 Gsample/s 10-tap full-rate variable-latency self-timed FIR filter with clocked interfaces," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2002, vol. 1, p. 60.

[17] N. Sayiner, H. V. Sorensen, and T. R. Viswanathan, "A level-crossing sampling scheme for A/D conversion," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 43, no. 4, pp. 335–339, Apr. 1996.

[18] M. Bazes, "Two novel fully complementary self-biased CMOS differential amplifiers," *IEEE J. Solid-State Circuits*, vol. 26, no. 2, pp. 165–168, Feb. 1991.

[19] G. Gerosa, S. Gary, C. Dietz, D. Pham, K. Hoover, J. Alvarez, H. Sanchez, P. Ippolito, T. Ngo, S. Litch, J. Eno, J. Golab, N. Vanderschaaf, and J. Kahle, "A 2.2 W, 80 MHz superscalar RISC microprocessor," *IEEE J. Solid-State Circuits*, vol. 29, no. 12, pp. 1440–1454, Dec. 1994.

[20] I. Sutherland, R. F. Sproull, and D. Harris, *Logical Effort: Designing Fast CMOS Circuits*, 1st ed. San Mateo, CA: Morgan Kaufmann, 1999.

[21] W. J. Dally and J. W. Poulton, *Digital Systems Engineering*, 1st ed. Cambridge, U.K.: Cambridge Univ. Press, 1998.

[22] T. Miki, Y. Nakamura, M. Nakaya, S. Asai, Y. Akasaka, and Y. Horiba, "An 80-MHz 8-bit CMOS D/A converter," *IEEE J. Solid-State Circuits*, vol. SC-21, no. 6, pp. 983–988, Dec. 1986.

[23] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-Time Signal Processing*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 1999.

Yee William Li (S'00–M'05) received the B.Eng. degree (First Class Honors) in computer engineering from The University of Hong Kong, Pokfulam, Hong Kong, and the M.S., M.Phil., and Ph.D. degrees in electrical engineering from Columbia University, New York, NY.

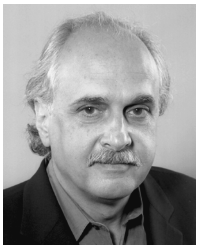
He was with Motorola Semiconductors and IBM T. J. Watson Research Center and served as a Visiting Lecturer with The University of Hong Kong. He is currently with the Advanced Design Mixed-Signal Circuit Group, Intel Corporation, Hillsboro, OR. His research focuses on low-power circuit design techniques in leading-edge CMOS technologies, including PLLs, delta–sigma modulators, on-chip dc–dc converters, and thermal sensors design.

Dr. Li was a recipient of the Croucher Foundation Scholarship. While he was with Motorola semiconductors, he received the Engineering Excellence Award. He received the Best Paper runner-up Award at the IEEE International Symposium on Asynchronous Circuits and Systems and a winning entry in the Design Contest of International Symposium on Low Power Electronics and Design.

Kenneth L. Shepard (S'85–M'92–SM'03) received the B.S.E. degree from Princeton University, Princeton, NJ, in 1987, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, in 1988 and 1992, respectively.

From 1992 to 1997, he was a Research Staff Member and Manager with the VLSI Design Department, IBM T. J. Watson Research Center, Yorktown Heights, NY, where he was responsible for the design methodology for IBM's G4 S/390 microprocessors. Since 1997, he has been with Columbia University, where he is now an Associate Professor. He also served as Chief Technology Officer of CadMOS Design Technology, San Jose, CA, until its acquisition by Cadence Design Systems in 2001. His current research interests include design tools for advanced CMOS technology, on-chip test and measurement circuitry, low-power design techniques for digital signal processing, low-power intrachip communications, and CMOS imaging applied to biological applications.

Dr. Shepard was the recipient of the Fannie and John Hertz Foundation Doctoral Thesis Prize in 1992. While at IBM, he received Research Division Awards in 1995 and 1997. He was also the recipient of a National Science Foundation CAREER Award in 1998 and IBM University Partnership Awards from 1998 through 2002. He was also awarded the 1999 Distinguished Faculty Teaching Award from the Columbia Engineering School Alumni Association. He has been an Associate Editor of the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS and was the technical program chair and general chair for the 2002 and 2003 International Conference on Computer Design, respectively. He has served on the program committees for ICCAD, ISCAS, ISQED, GLS-VLSI, TAU, and ICCD.



Yannis P. Tsividis (S'71–M'74–SM'81–F'86) received the B.S. degree from the University of Minnesota, Minneapolis, in 1972, and the M.S. and Ph.D. degrees from the University of California, Berkeley, in 1973, and 1976, respectively.

He is the Charles Batchelor Memorial Professor of Electrical Engineering with Columbia University, New York, NY. He has been with Motorola Semiconductor and AT&T Bell Laboratories and has taught at the University of California at Berkeley, the Massachusetts Institute of Technology, Cambridge, and the National Technical University of Athens, Greece. His latest book is *Operation and Modeling of the MOS Transistor* (Oxford University Press, 2003, second edition).

Prof. Tsividis was the recipient of the 1984 IEEE Baker Best Paper Award, the 1986 European Solid-State Circuits Conference Best Paper Award, and the 1998 IEEE Circuits and Systems Society Guillemin–Cauer Best Paper Award. He was a corecipient of the 1987 IEEE Circuits and Systems Society Darlington Best Paper Award and the 2003 International Solid-State Circuits Conference L. Winner Outstanding Paper Award. He received the Presidential Teaching Award from Columbia University in 2003 and the IEEE Undergraduate Teaching Award in 2005.