

# An 8-Bit 150-MHz CMOS A/D Converter

Yun-Ti Wang and Behzad Razavi  
Electrical Engineering Department  
University of California, Los Angeles

## Abstract

An 8-bit, 5-stage pipelined ADC employs sliding interpolation, reinterpolation, and interleaving with clock edge reassignment. Fabricated in a 0.6- $\mu\text{m}$  CMOS technology, the ADC achieves a DNL of 0.62 LSB, INL of 1.24 LSB, and SNDR of 43.7 dB at 150 MHz sampling rate. The converter draws 395 mW from a 3.3-V supply and occupies an area of 1.2 x 1.5 mm<sup>2</sup>.

## I. INTRODUCTION

High-speed analog-to-digital converters (ADCs) find wide application in instrumentation and communication systems. The design of 8-bit converters operating above 100 MHz presents many challenges, especially in CMOS technology.

This paper describes the design of an 8-bit 150-MHz ADC implemented in a 0.6- $\mu\text{m}$  CMOS process. With a 6.6-ns clock period and transition times on the order of 0.5 ns, the timing budget for both sampling and quantization is extremely tight, prohibiting the use of high-precision feedback stages for signal processing.

This work introduces a number of architecture and circuit techniques that obviate the need for closed-loop circuits in multi-stage A/D converters, thereby relaxing the speed-precision trade-off. Using such techniques and an open-loop front-end sample-and-hold amplifier (SHA), the converter performs pipelining with no interstage D/A converters, subtractors, or precision charge transfer.

## II. ARCHITECTURE AND CIRCUIT DESIGN

The converter architecture is shown in Fig. 1. The front-end SHA is followed by stage 1, which consists of 17 differential pairs and a 4-bit coarse A/D converter. Each of stages 2 through 5 performs sliding and multiplexing (described below), distributed sampling, 2x interpolation, and a 2-bit A/D conversion. An overall redundancy of 4 bits relaxes the precision required of the sub-ADCs.

### A. Sliding Interpolation

We now present the concept of sliding interpolation. Consider the simple 2x interpolation network shown in Fig.2(a). While this scheme reduces the number of the input preamplifiers and hence the input capacitance, it nonetheless requires a large number of differential pairs and comparators. How-

ever, we recognize that for a given input level, the outputs of

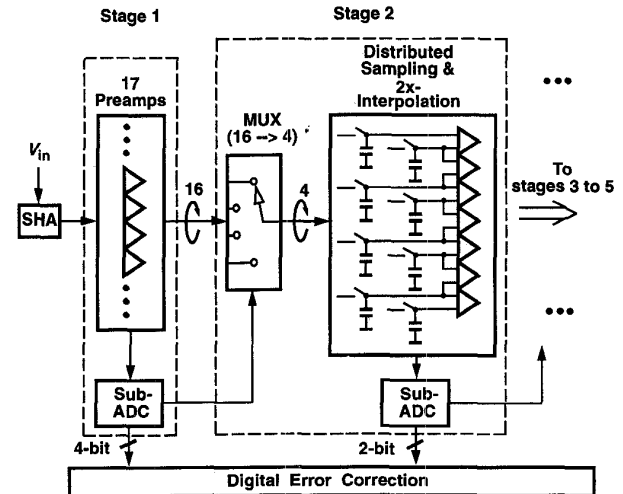


Fig. 1: ADC architecture.

only a few preamplifiers in the first stage are of interest. Thus, the subsequent stages need not interpolate the outputs of *all*

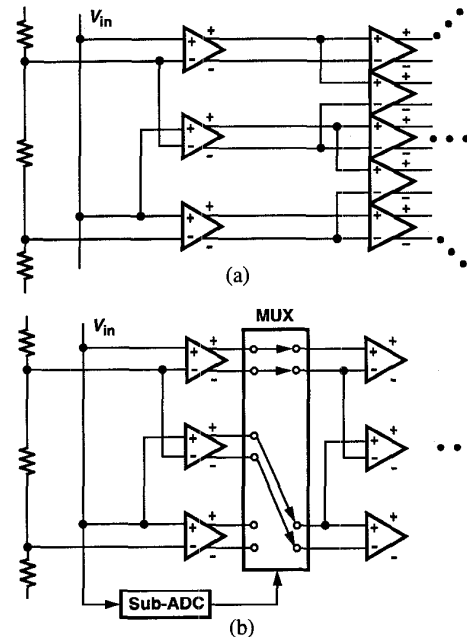


Fig. 2: Interpolation architectures, (a) traditional, (b) sliding.

of the preamplifiers. We then surmise that a compact interpolating stage can “slide” up and down if the analog input value is roughly known. Shown in Fig. 2(b), the idea is to use a coarse ADC to determine which preamplifier outputs must be interpolated, and route those outputs to the interpolating differential pairs. While, in principle, only two outputs are sufficient, in this design we interpolate between 4 outputs to allow margin for offsets. By virtue of this technique, the number of interpolating differential pairs drops from roughly 500 to 50. The five sub-ADCs consume a total of 28 simple comparators.

As shown in Fig.1, the interstage sampling is accomplished by a distributed sampling network similar to that reported in [1]. The interface between the multiplexer and the interpolation stage provides a natural point for sampling because the long interconnect lines required at the output of the multiplexer can serve as sampling capacitors.

Sliding interpolation offers a number of benefits. First, it easily lends itself to multi-stage pipelining with no D/A converters or subtractors. Second, it requires *no* precision gain in any of the building blocks, allowing the use of simple differential pairs in the entire signal path. Third, it can include reinterpolation to improve the precision.

### B. Reinterpolation

An important benefit of interpolation is the reduction in differential nonlinearity (DNL) [2,3] that would result from the offset of the preamplifiers. However, integral nonlinearity (INL) still remains uncorrected, demanding large input devices. To alleviate the problem, a “reinterpolation” method is used here. As depicted in Fig. 3, if only the *interpolated* zero crossings are sensed by the following stages and the original zero crossings are discarded, the INL is reduced by 40%; Monte Carlo simulations confirm this result. Note, that this redundancy is necessary only in the first stage of the pipeline, where the cumulative gain is still low; in stages 2 through 5 all zero crossings are utilized.

### C. Interleaving

Even though the maximum path “length” between consecutive samplers in the pipeline corresponds to roughly two differential pairs, the settling requirements still limit the conversion speed. For this reason, the converter employs the interleaving scheme illustrated in Fig. 4, where the first stage preamplifiers and all of the sub-ADCs are shared between the two paths. The timing is such that, when one stage in the odd channel is in the sampling mode, the corresponding stage in the even channel is in the hold mode and vice versa.

Fig. 4 also shows an additional “replica” front-end SHA, whose output is quantized by the first sub-ADC. Scaled down in device size and current levels by a factor of two with respect to the main SHA, the replica prohibits the large kick-back noise of the sub-ADC from corrupting the output of the main SHA.

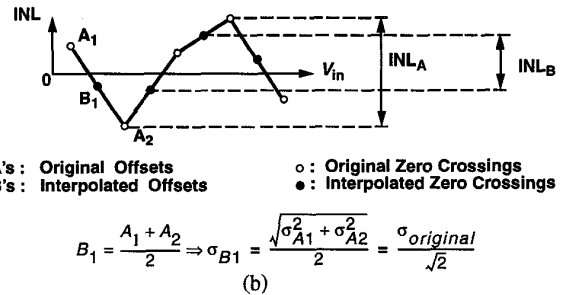
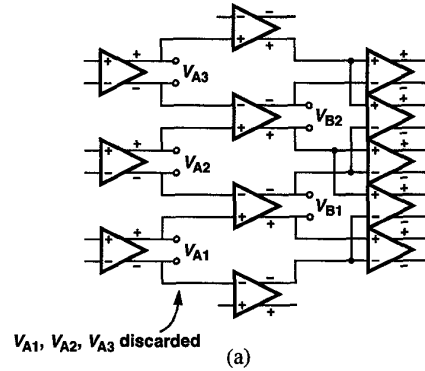


Fig. 3: Reinterpolation, (a) implementation, (b) error plot.

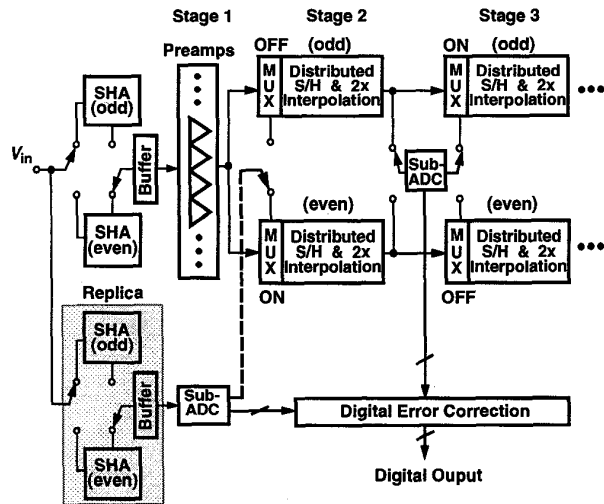


Fig. 4: Pipelining with interleaving architecture.

### D. Clock Edge Reassignment

The interleaving of the front-end SHAs must deal with the problem of gain and timing mismatch. While gain mismatch can be reduced to the 8-bit level by proper sizing, the timing mismatch between the two clock phases that alternately drive the odd and even SHAs becomes problematic. This issue is alleviated through the use of “clock edge reassignment.” Illustrated in Fig. 5, the idea is to utilize the very same clock edge to define the sampling point in both SHAs. Here, two

“predictive” waveforms  $V_{odd}$  and  $V_{even}$  enable one of the switches, routing the falling edge of the master clock to either

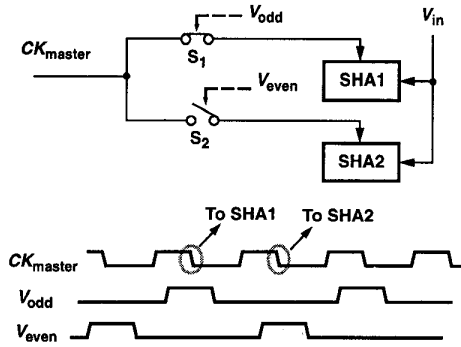


Fig. 5: Concept of the clock edge reassignment.

of the SHAs. The timing mismatch now corresponds to the propagation delay mismatch between  $S_1$  and  $S_2$ , and two switches inside SHA1, and SHA2, which can be maintained well below 10 picoseconds.

#### D. Building Blocks

The implementation of the front-end SHA plays a critical role in the dynamic behavior of the converter. In order to achieve fast settling, the circuit uses simple top-plate sampling and a PMOS source follower (Fig. 6). Simulations indi-

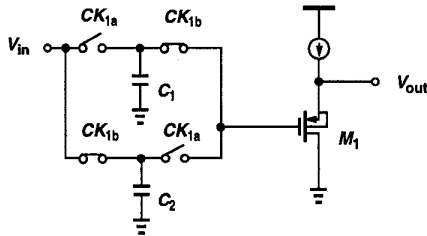


Fig. 6: Simplified diagram of dual-channel interleaved SHA.

cate that two PMOS source followers operating differentially with their sources connected to their n-wells achieve about 10 bits of linearity. The nonlinearity due to charge injection is lowered to below the 8-bit level by differential operation as well as large sampling capacitors (1 pF). Note that only the sampling capacitors are interleaved whereas the follower is shared, thereby reducing the gain mismatch. The finite input capacitance of the follower also results in an equivalent resistance connected between the outputs of the two channels, in essence yielding a gain roll-off at high frequencies. With proper design, this roll-off is limited to 1 dB at 75 MHz.

In the actual design, the front-end SHA is realized with *triple-channel* interleaving. This is because the sampling phase is quite faster than the hold/quantization/multiplex phase, thereby requiring a clock duty cycle of about 30% [Fig. 7(a)]. Since the duty cycle deviates substantially from 50%, it is difficult to employ dual-channel interleaving without any “dead” time. To resolve this issue, the clock period is divided into three equal time slots: one for front-end sam-

pling, one for sub-ADC, and one for multiplexing [Fig. 7(b)].

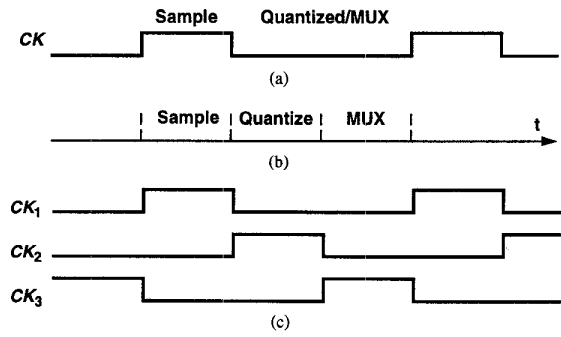


Fig. 7: Timing diagram for SHA.

The timing diagram of Fig. 7(c) is then used to interleave three sampling capacitors. (To generate the time slots with reasonable accuracy, the 150-MHz clock is divided by 3 on the chip.) The actual circuit is shown in Fig. 8.

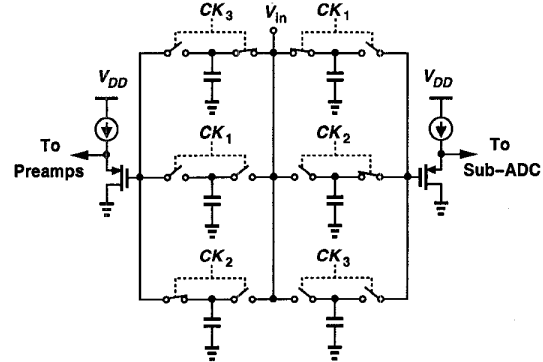


Fig. 8: Triple-channel interleaved SHA.

Fig. 9 shows the realization of a slice of the signal path. While interpolation by a factor of 2 tolerates large nonlinearity in differential pairs, the reinterpolation scheme does

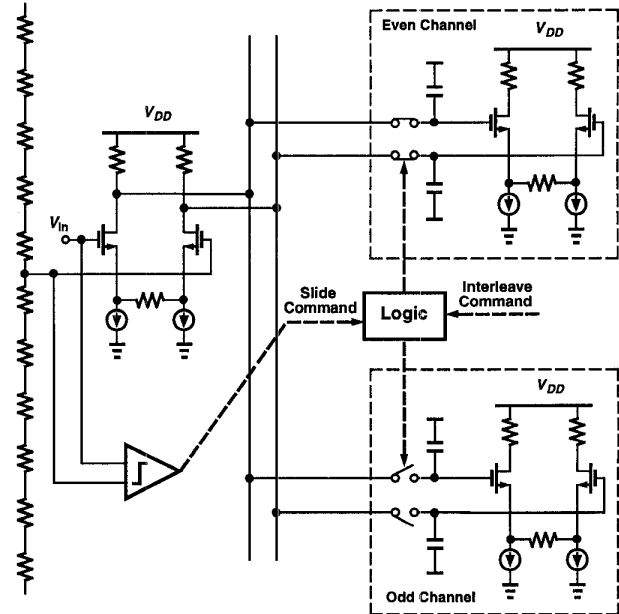


Fig. 9: Realization of a slice of the signal path.

require some linearity. Hence, the differential amplifiers in the signal path employ resistive degeneration. The actual design is fully differential.

It is also important to note that the converter requires no floating capacitors and can therefore utilize native metal-sandwich structures in digital CMOS technologies.

### III. EXPERIMENT RESULTS

The ADC has been fabricated in a 0.6- $\mu\text{m}$  CMOS technology, occupying an area of 1.2 mm x 1.5 mm with the active area of 1.2 mm<sup>2</sup>. The circuit is tested with a 3.3-V supply with differential input swings of 1.6 V<sub>pp</sub> and a sampling rate of 150 MHz.

Fig. 10 shows the measured DNL and INL profiles obtained from code density tests. The maximum values of DNL and INL are 0.61 and 1.24 LSB, respectively. The

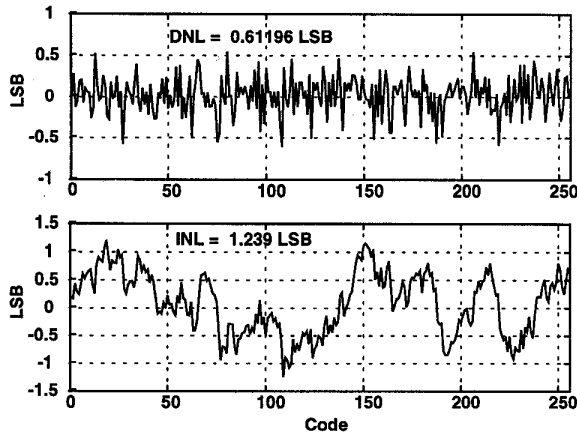


Fig. 10: DNL and INL @  $f_{in} = 1.8$  MHz.

dynamic performance of the converter is measured in the frequency domain. Fig. 11 depicts the spectrum of the recon-

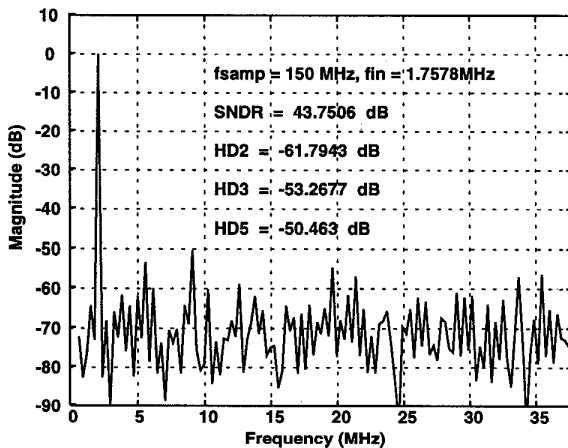


Fig. 11: FFT @  $f_{in} = 1.76$  MHz.

structed signal at 1.76 MHz, exhibiting harmonics 50 dB below the fundamental and a signal-to-(noise+distortion)

ratio (SNDR) of 43.7 dB. The spurious-free dynamic range (SFDR) and SNDR as a function of the analog input frequency are plotted in Fig. 12. Table 1 summarizes the overall performance.

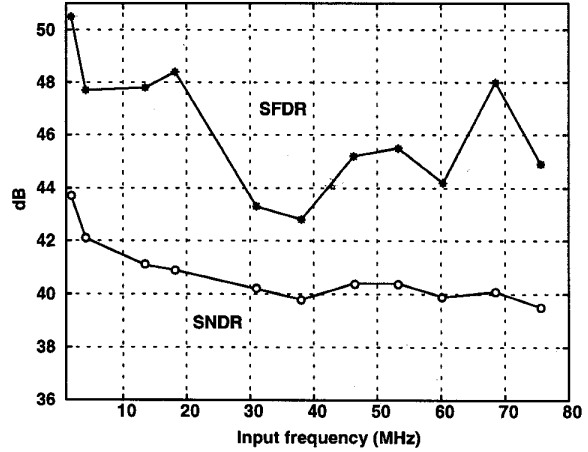


Fig. 12: SNDR and SFDR @ sampling rate = 150 MHz.

Technology	0.6- $\mu\text{m}$ , 1-poly, 3-metal CMOS
Resolution	8 bits
DNL	0.62 LSB
INL	1.24 LSB
Sampling Rate	150 MHz
SNDR @ $f_{in}=1.8$ MHz	43.7 dB
$f_{in}=70$ MHz	40 dB
Analog Input Swing	1.6 V <sub>p-p</sub>
Input Capacitance	1.5 pF
Active Chip Area	1.2 mm <sup>2</sup>
Supply Voltage	3.3 V
Power Consumption	
Analog	330 mW
Digital	53 mW
Reference Ladder	12 mW
Total	395 mW

Table 1: Measurement Summary.

### References

- [1] A. G. W. Venes, R. J. van de Plassche, "An 80-MHz, 80-mW, 8-b CMOS Folding A/D Converter with Distributed Track-and-Hold Preprocessing," *IEEE J. Solid-State Circuits*, vol. SC-31, pp. 1846-1853, Dec. 1996.
- [2] C. Lane, "A 10-Bit, 60-MSPS Flash ADC," *Proc. BCTM*, pp. 44-47, Sep. 1989.
- [3] H. Kimura et al., "A 10-b 300-MHz Interpolated Parallel A/D Converter," *IEEE J. Solid-State Circuits*, vol. SC-28, pp. 438-446, Apr. 1993.