

ON-LINE HISTOGRAM EQUALIZATION FOR FLASH ADC

Yanyi L. Wong, Marc Cohen and Pamela Abshire

ECE Dept., University of Maryland College Park, MD 20742 Email: yanyi.wong@ieee.org

Abstract—We present theory, design and measurement results for an on-line histogram equalization algorithm implemented on a 750MS/s 6b flash analog to digital converter in standard $0.35\mu\text{m}$ CMOS. The user simply turns on "training mode" for a few seconds, while the algorithm automatically adjusts comparator levels to match the observed input signal distribution. This results in signal conversion with equal probability for each of the output codewords. The new architecture is an extension of a flash ADC incorporating an adaptive floating gate comparator and control circuits for automatic programming of the reference levels. Experiments show output codes with at least 5.9 bits entropy for ramp, sine and Gaussian random signals after adaptation. Uniform programming produces 5.7 ENOB for input frequencies up to 200MHz and maximum DNL and INL of 0.24 LSB and 0.79 LSB at Nyquist rate, while equalization produces 5.3 ENOB up to 600MHz.

I. INTRODUCTION

Analog-to-digital (AD) conversion is a mapping from ranges of continuous values to discrete values. Flash AD converters (ADC) are an architecture in which every comparator has a reference value that explicitly defines the boundary for AD mapping. Typical flash ADCs use equally-spaced values for a linear AD mapping.

Non-linear AD mapping can sometimes outperform linear mapping for a given signal distribution. Examples include companding algorithms such as A-law and μ -law widely used in audio communications [1]. Max [2] and Lloyd [3] provided algorithms for optimal conversion for a signal with known probability density function.

Since each reference level t_n is explicitly defined in a flash ADC, it is a natural architecture for non-linear signal conversion. Hasler et al. previously reported flash ADCs using programmable e-pots as individual comparators that define t_n 's [4], [5]. After a suitable mapping profile is calculated, the user trims each e-pot and monitors adjustments before use.

We have developed an on-chip, automatic histogram equalization algorithm which has been implemented in a proof-of-concept 750MS/s 6b flash ADC. The algorithm uses the self-adaptive feature in a previously designed comparator [6], and extends the adaptation to match time-varying input signals. Once trained, each output codeword is produced with equal probability, and consequently the histogram for codewords is equalized. In a typical experiment, the user simply turns on training mode for a few seconds, after which a nearly uniform output code histogram is observed. Since the conversion is usually non-uniform, the analog values must be recovered with a specific digital-to-analog (DA) mapping $T(n)$. In this paper, we will discuss the role of $T(n)$, present a method for obtaining $T(n)$, and describe details including the histogram equalization algorithm, implementation, circuitry and measured performance.

II. NON-LINEAR AD CONVERSION

Figure 1 depicts 6-bit non-linear AD conversions for (a) sine and (b) Gaussian-distributed signals. A continuous-time sine signal spends more time near the maximum and minimum values than the middle, but a Gaussian-distributed signal spends most of the time near the middle value. $F_1(v) = P[X < v]$ denotes the cumulative distribution function (cdf) for input random variable X ; $T(n) : n \rightarrow v$

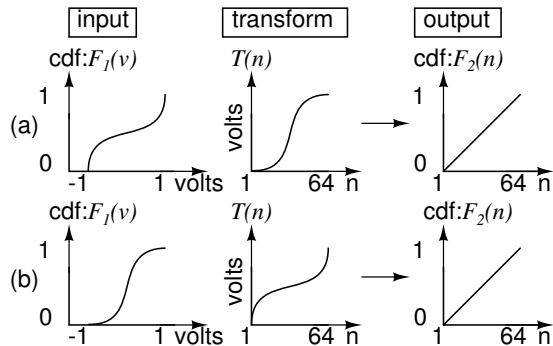


Fig. 1. A non-linear transfer function T can equalize a non-uniformly distributed input F_1 .

the DA mapping function for 64 discrete n values, and $F_2(n)$ the output cdf. By substituting v with $T(n)$ in $F_1(v)$ we obtain $F_2(n)$; $F_1(T(n)) = F_2(n)$. By taking inverse of F_1 on both sides, we can easily find the DA mapping function

$$T(n) = F_1^{-1}(F_2(n)) \quad (1)$$

Therefore, the input and output cdf together determine the DA mapping useful for recovering the analog signal. In the case of histogram equalization, the desired probability mass function (pmf) is uniform and the desired output cdf is linear as shown in Fig.1, so $T(\cdot)$ is simply the inverse of $F_1(\cdot)$.

In an N -bit flash ADC, we have 2^N and $2^N - 1$ discrete values for the DA mapping $T(n)$ and the comparator reference levels t_n , respectively. Once $T(n)$ is obtained using (1), it is usually adequate to assume that $t_n = (T(n) + T(n+1))/2$, for $n=1$ to $2^N - 1$. The ADC outputs code n if the input X falls between t_{n-1} and t_n , for n from 1 to 2^N , where $t_0 \equiv -\infty$ and $t_{2^N} \equiv +\infty$.

III. HISTOGRAM EQUALIZATION

Figure 2 demonstrates histogram equalization in a 3-bit flash converter. Suppose the input signal has the pdf shown in (a), the cdf $F_1(v)$ in (b), and undergoes non-linear AD conversion with the t_n 's marked on the x-axis of (b). The resulting output probability mass function (pmf) is uniform as shown in (c). Code 1 occurs with probability $F_1(t_1)$, code 2 with $F_1(t_2) - F_1(t_1)$, and so on. Clearly histogram equalization implies that $F_1(t_n) - F_1(t_{n-1}) = 1/8$, for $n = 1, 2, \dots, 8$, where $t_0 \equiv -\infty$ and $t_8 \equiv +\infty$.

For implementation the algorithm has been structured according to equal-partitioning for reasons discussed in the next section. Level t_4 represents the middle comparator, which is responsible for the most significant bit (MSB) in the digital output, and we classify it as partition hierarchy 1. Level t_4 partitions the set of input signals into equally probable halves. At the next partition hierarchy, t_2 and t_6 divide the remaining partitions in halves, and so on.

For an N -bit flash ADC, with an input signal cdf $F_1(v)$, we achieve output code histogram equalization if we assign the t_n values such

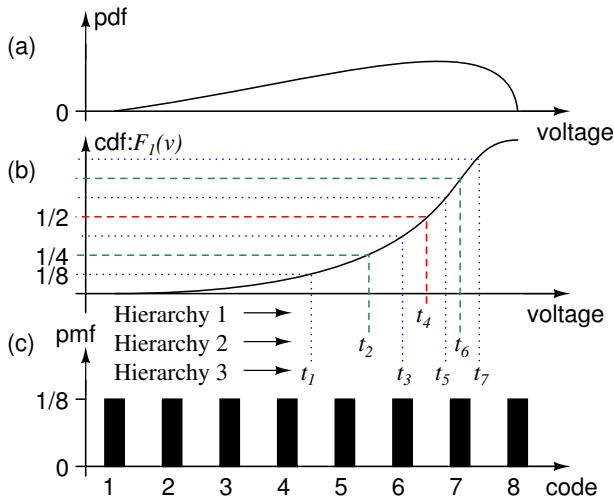


Fig. 2. The non-linear conversion equalizes histogram from analog input distribution (a) to digital output distribution (c).

that

$$F_1(t_n) - F_1(t_{n-1}) = 1/2^N, \quad (2)$$

for $n = 1, 2, \dots, 2^N$, where $t_0 \equiv -\infty$ and $t_{2^N} \equiv +\infty$. It is easy to extend this to an equal-partitioning algorithm using the conditional cdf: $F_1(v|A) = P[(X < v) \cap A]/P[A]$. Summing (2) over n , we have a single partition at hierarchy 1: $\sum_{n=1}^{2^N-1} (F_1(t_n) - F_1(t_{n-1})) = F_1(t_{2^N-1}) = 1/2$. Similarly, we have two partitions at hierarchy 2: $F_1(t_{2^N-2}|X < t_{2^N-1}) = 1/2$ and $F_1(t_{2^N-2+2^N-1}|X > t_{2^N-1}) = 1/2$. In general, for hierarchy l , the partitions resulting from the previous levels $1, 2, \dots, l-1$ are again partitioned into halves: $F_1(t_{2^N-l+(n-1)2^{N-l+1}}|t_{(n-1)2^{N-l+1}} < X < t_{n2^{N-l+1}}) = 1/2$, for $n = 1, 2, 3, \dots, 2^{l-1}$. In this way, an update direction for each reference level t_n is determined by monitoring the corresponding conditional cdf, determined from the appropriate subset of all reference levels. If t_n is too high, the conditional cdf is greater than $1/2$, and if t_n is too low, the conditional cdf is less than $1/2$. Therefore the update law for the reference level t_n is specified by:

$$\Delta t_n = \alpha_n \text{Sign} \left(F_1(t_n|A_n) - \frac{1}{2} \right)$$

where A_n is the range corresponding to the partitioning operation of reference level t_n .

We simulated this algorithm for a 4-bit flash ADC; the 15 reference levels and output code entropy are plotted against sample number in Fig.3 (a) and (b), respectively. The entropy $H = -\sum p \log_2 p$ is a good indication of the flatness of the output histogram. For an equalized histogram, the entropy is equal to the number of bits.

The 15 reference levels were initialized to random values, simulating typical initial floating gate states after manufacturing. As the t_n 's gradually converge to their final positions, the entropy steadily rises to 4 bits. The t_n 's quickly track the input, which changes from uniform to Gaussian distribution at the 128kth sample and subsequently to exponential distribution at the 192kth sample. The uniform signal is distributed between +1V and -1V, the Gaussian signal has zero mean with $\sigma = 0.33V$, and the exponential signal has mean 0.4V and offset -1V. We set the update increment (α_i) to 37.5 μV per sample for partition hierarchy 1. Since hierarchy 2 is updated half as frequently as hierarchy 1, the adjustment increments for these levels is set to 75 μV . Hierarchies 3 and 4 are set to 0.15mV and 0.3mV, respectively. For higher adjustments we observed faster convergence

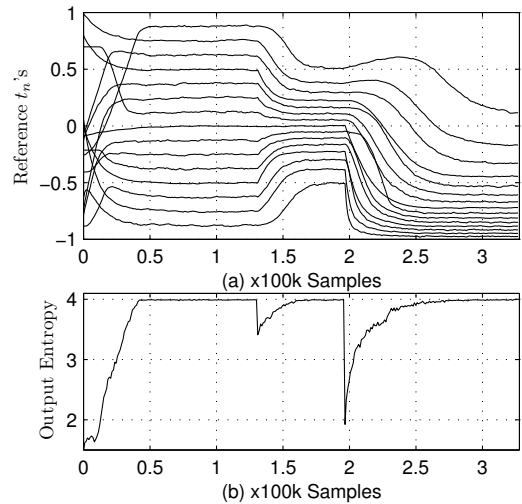


Fig. 3. The reference levels t_n 's in a 4-bit flash ADC adapt as the input distribution change from uniform, Gaussian to exponential.

but coarser t_n values; for lower adjustments we observed slower convergence but finer values.

IV. IMPLEMENTING HISTOGRAM EQUALIZATION IN 6B FLASH

We use a previously developed adaptive floating gate comparator (AFGC) [6] as the basic element for on-line histogram equalization. Figure 4 (a) shows the block diagram for the AFGC. It receives differential analog inputs $V_{i+,-}$ and produces complementary output bits D and \bar{D} . The AFGC is capable of adapting its reference level t to a DC value X supplied at the input $V_i = V_{i+} - V_{i-}$ by adjusting its internal offset V_O . From the law of large numbers, it is easy to show that for a random variable X as input, the reference level t will be adapted to the mean of X (i.e., $t \rightarrow E[X]$), if X is stationary, the adaptation is carried out many times, and the increment Δt for each adaptation is small.

The above feature directly fulfills the equal-partition algorithm for the first hierarchy: $F_1(t) = F_1(E[X]) = 1/2$. For subsequent hierarchies, the update is conditioned according to the partition in which a particular sample falls. In Fig.4 (a), terminal P is used to enable adaptation. Thus, we realize conditioning by controlling P to enable or disable adaptation. Figure 4 (b) shows the algorithm for an example 3-bit ADC. Each AFGC performs adaptation only if the input falls within its corresponding adaptation range.

V. CIRCUITRY

Figure 5 shows the AFGC circuit. In previous work [6], both inputs had floating gates and update was accomplished using hot electron injection only. Here, the floating gate is on the positive input side only, and both hot-electron injection [7] and tunneling [8] are used to adjust the floating gate voltage. The AFGC uses the comparison outcome for adjusting its input offset in small increments, and accomplishes offset adaptation over many adjustments. Each AFGC consists of one comparator (CP1), two charge pumps and control logic, and stores its offset V_O on capacitor C_1 , which couples the floating gate V_{fg} to the multiplexer MX1. V_{fg} is connected to the positive input of CP1, whose negative input is connected to V_{i-} via dummy devices used for symmetry. The programming enable signal (PE) determines the operational mode. During operation (PE=0), MX1 passes the positive input signal V_{i+} to the comparator via C_1 . The comparator passes its result to the encoder. During programming (PE=1), MX1 forms a feedback loop for OP-amp A1, configured

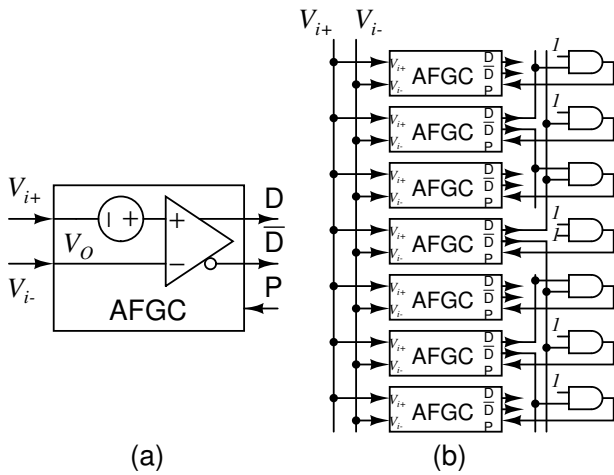


Fig. 4. (a) The AFGC is capable of adapting the internal reference values to the input. (b) The histogram equalization algorithm is implemented with AFGCs with digital AND gates.

as a voltage follower. Regardless of the offset or input values, A1 sets V_{fg} close to an externally supplied reference voltage V_{ref} so that programming charges can be applied in controlled increments. Data hold signal (DH) is asserted one clock cycle ahead of PE, and the most recent comparison outcome before entering programming is written to a register (REG). An outcome of LO means that offset V_O is too low, and tunneling is activated to raise it. Conversely, an outcome of HI means that V_O is too high, and hot-electron injection is activated to reduce it. The AFGC quickly reaches a steady state wherein V_O is close to desired offset.

The dashed circles mark the tunneling and injection sites. The tunneling site is the gate oxide of a pFET with source and drain shorted to the nWell (V_i). The injection site is the gate oxide of a pFET (M1), whose drain V_i is pulsed below GND to induce a high electric field in the channel. M2 sets the channel current for M1. High voltage buffers HB1 and HB2 generate the short voltage pulses required during programming. V_i is driven by HB1 to roughly 8V for tunneling and V_i is driven by HB2 to $-2V$ for injection. Bias voltages and currents are adjusted to induce $\pm 1V/sec$ change on V_O via tunneling or injection. For a 1V peak-to-peak signal range, the maximum required change in V_O is 1V, so programming time is set to 1.2s for each AFGC to ensure complete programming.

In the 6-bit ADC, the analog input is first sampled by a source-follower track-and-hold circuit (T/H), and then quantized by 63 AFGCs. Resulting thermometer codes are converted to quasi-gray codes and finally binary codes. The designs for T/H and comparator follow [9]. We use complementary bit lines and sense-amps to ensure code conversion speed and correctness, true single-phase clocked (TSPC) D-flip-flops [10] for all synchronous logic, and large inverters to buffer the clock signal. Output buffers are designed to drive 75Ω transmission lines. They deliver digital outputs along with the clock signal to drive a logic analyzer synchronously (in state mode). Both clock and input terminals are terminated with 50Ω resistors on-chip. A control logic block generates required timing signals for each AFGC during programming. The user can override the algorithm and program the t_n 's one-by-one with preset DC values supplied to the input.

VI. MEASUREMENT AND RESULTS

A 1GS/s arbitrary waveform generator (AWG) is used to supply arbitrary input signals. Two RF signal generators sharing the same

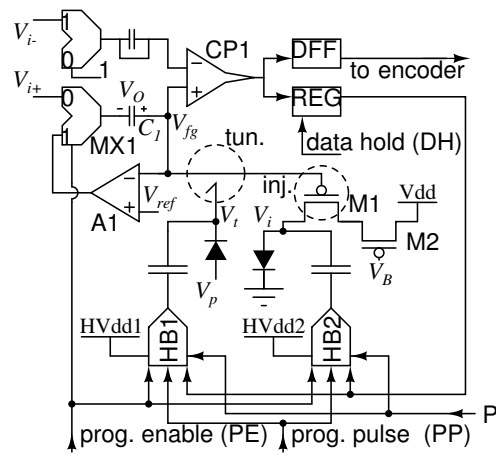


Fig. 5. Each AFGC comprises an offset storing capacitor C_1 , a comparator CP1, floating gate V_{fg} , tunneling/injection sites, charge pumps and high voltage buffers.

10MHz reference are used to provide sine waves for both input and clock signals for standard DSP-based coherent single tone analysis [11]. An 800MS/s logic analyzer is used to capture digital output. A phase splitter and bias-T's are used to generate differential input for the sine wave. On-chip inverters generate non-overlapping digital clock signals from the sine wave clock input. Typical power consumption is 1.1W at 3.3V supply and an operating frequency of 750MHz, of which 60% is used in the clock buffer inverters. The 3mm x 4.5mm chip is attached to a copper plate and bonded directly onto a 4-layer PCB, enclosed with thermally conductive epoxy. The PCB is attached to a thermoelectric plate, a heat sink and a fan, and the ADC is fully functional with the surface temperature from $9^\circ C$ to $23^\circ C$ during testing. We use both the AWG and the signal generator to provide signals of known distribution for calibration. In practice a calibration signal source can be incorporated on the PCB.

We matched the injection and tunneling rates to 0.6V/sec for all hierarchies. We first program the t_n 's manually to uniformly distributed values between 0.5V to $-0.5V$ using the AWG, then use a 3dBm sine wave as input. We analyze 16k captured samples in spectral domain using Fast Fourier Transform. A typical plot of the spectrum is shown in Fig.6(a). The signal to noise-plus-distortion ratio (SNDR) is the ratio of signal power at fundamental frequency to the sum of all other power excluding DC. The effective number of bits (ENOB) is calculated as $ENOB = 6.02^{-1}(SNDR - 1.76)$. The ENOB is plotted versus input signal frequency in Fig.6(b) as circles. The sine power is adjusted to obtain full scale in the digital codes. Next, we perform adaptation for input signals at each frequency with 2dBm input power, capture and decode the data using Eqn.(1) and perform analysis. The ENOB after adaptation is plotted as dots in (b). Figure 6(c) shows the output code histograms for both manual programming and adaptation for 387MHz sine wave input. The DNL and INL for manual programming are calculated from the histogram to be 0.24 and 0.79 LSB, respectively.

Next, we perform adaptation with 3 different signals at input: a) 1.54MHz, 1V peak-to-peak triangular wave, b) 3dBm 387MHz sine wave and c) a Gaussian random signal with zero mean, $\sigma = 166mV$ at a data rate of 375MS/s. a) and c) are generated using the AWG. We normalize the histogram to obtain the pmf and subsequently the cdf. The resulting cdfs are plotted in Fig.7. The slight deviation from an ideal cdf can be attributed to offset in the op-amp, mismatch of the injection and tunneling currents, and mismatch in charge pumps

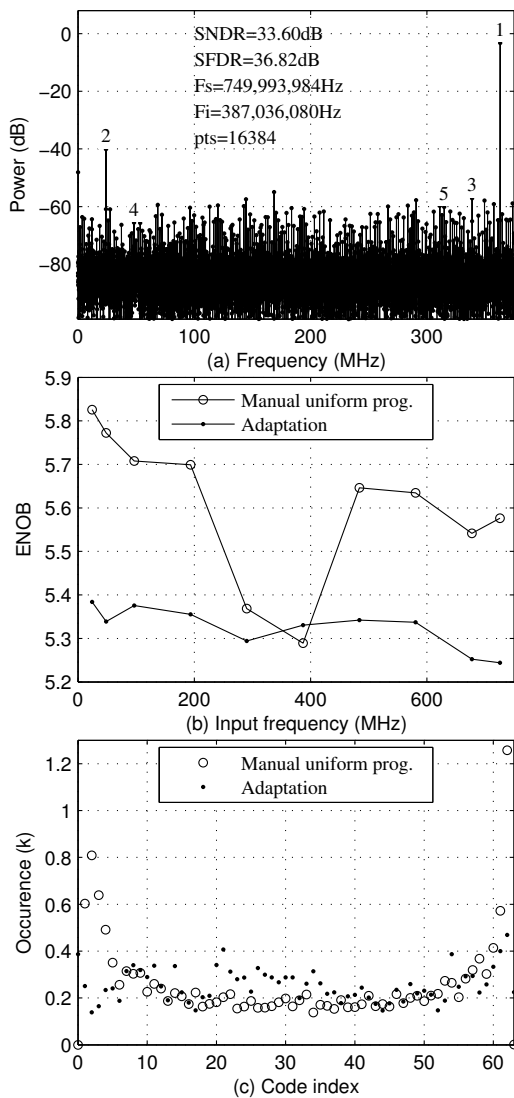


Fig. 6. Performance results for adaptive flash ADC: (a) SNDR computed from 16k digital output samples, (b) ENOB vs. input frequency and (c) histogram.

in each AFGC. The calculated entropies are 5.96, 5.95, and 5.90 bits, and the maximum DNL are 0.62, 0.83 and 1.75 LSB for a), b) and c), respectively. For comparison, we program the ADC manually with uniformly distributed t_n 's and observed entropy of 5.99, 5.73, and 5.34 bits for a), b) and c), respectively. Adaptation to periodic signals as in a) and b) requires careful selection of the input frequency such that sufficiently distinct values are sampled for adaptation. We have chosen an input frequency such that 256 distinct values in each period of the signal are used for adaptation.

We monitored continuous operation for one month, and observed sporadic bit errors with an error rate of 2.7×10^{-9} with no sign of floating gate charge leakage. We monitor bit error rate by sampling a 0.97MHz full scale sine wave at 750MS/s, and from a sequence of 1M samples count each instance of two or more LSB changes in consecutive code transitions as an error. The experiment is repeated every 40 seconds. In one week we observed a total of 37 error bits.

VII. CONCLUSION

An algorithm for online histogram equalization was implemented by equal-partitioning of signal probability distribution in a hierarchi-

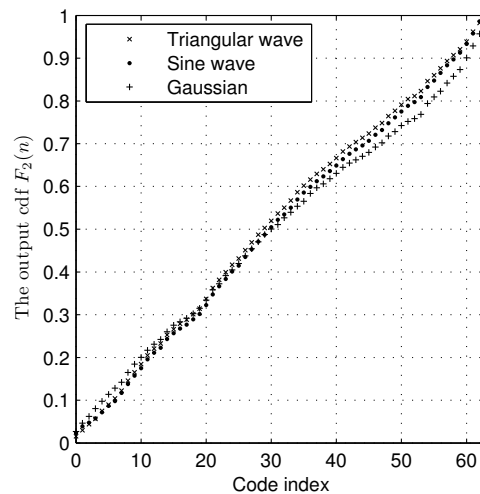


Fig. 7. Although output cdf deviates slightly from an ideal uniform cdf with residue norm of 0.13, the output cdf for three very different inputs are nearly identical.

cal fashion from the most significant bit to the least significant bit. The self-adaptive feature of a floating gate comparator matches the algorithm well when incorporated into a flash ADC. We adopt various high speed design methodologies to realize 750MS/s in standard $0.35\mu\text{m}$ technology. The flash ADC equalizes histogram for vastly varied signal distributions with entropy of 5.9 bits or higher. When properly decoded, the adapted flash ADC exhibits 5.3 ENOB up to 600MHz of input frequency. Histogram equalization is useful in situations where the signal distribution changes over time.

VIII. ACKNOWLEDGEMENT

This material is based on work supported by NSF under grant number 0238061. The authors thank Nicole Nelson and Seokjin Kim for assistance with wire-bonding and suggestions on DSP based test setup.

REFERENCES

- [1] *Federal Standard 1037C: Glossary of Telecommunications Terms*, The Institute for Telecommunication Sciences, Oct 2006, <http://www.its.bldrdoc.gov/fs-1037/fs-1037c.htm>.
- [2] J. Max, "Quantization for minimum distortion," *IRE Trans. Info. Theory*, vol. 6, no. 1, pp. 7–12, March 1960.
- [3] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Info. Theory*, vol. 28, no. 2, pp. 129–137, March 1982.
- [4] P. Brady and P. Hasler, "Offset compensation in flash ADCs using floating-gate circuits," in *Proc. IEEE ISCAS*, May 2005.
- [5] V. Krishnan, C. Duffy, D. Anderson, and P. Hasler, "Optimal quantization employing programmable flash analog to digital converters," in *Proc. 38th Asilomar Conf. Sig. Syst. Comp.*, vol. 1, 2004, pp. 816–819.
- [6] Y. Wong, M. Cohen, and P. Abshire, "A 1.2GHz adaptive floating gate comparator with 13-bit resolution," in *Proc. IEEE ISCAS*, May 2005, pp. 6146–6149.
- [7] W. D. Brown and J. E. Brewer, *Nonvolatile Semiconductor Memory Technology: A Comprehensive Guide to Understanding and Using NVSM Devices*. Piscataway, NJ: Wiley-IEEE Press, 1997.
- [8] M. Lenzlinger and E. H. Snow, "Fowler-Nordheim tunneling into thermally grown SiO_2 ," *J. App. Phys.*, vol. 40, no. 1, pp. 278–283, 1969.
- [9] M. Choi and A. Abidi, "A 6-b 1.3-Gsample/s A/D converter in $0.35\text{-}\mu\text{m}$ CMOS," *IEEE JSSC*, vol. 36, no. 12, pp. 1847–1858, December 2001.
- [10] J. Yuan and C. Svensson, "New single-clock CMOS latches and flipflops with improved speed and power savings," *IEEE JSSC*, vol. 32, no. 1, pp. 62–69, January 1997.
- [11] M. Burns and G. W. Roberts, *An introduction to mixed-signal IC test and measurement*. New York, New York: Oxford University Press, 2001.