An Architecture and An Algorithm for Fully Digital Correction of Monolithic Pipelined ADC's

Eric G. Soenen and Randall L. Geiger

***Abstract*—Accurate trimming of the analog circuitry in analog/digital converters beyond 12 b is difficult. An alternative approach allows a margin of errors on all analog components and compensates for it in the digital domain. This paper describes such a method for pipelined or cyclic converters. Unlike in sigma-delta converters, no over-sampling is required. A powerful identification algorithm determines a limited number of digital coefficients, that linearize the response. No external measurement hardware is needed. Based on the known performance of state-of-the-art analog blocks, linearity of 16 b at multi-Mhz sampling rates seems achievable.**

I. Introduction

THE new self-calibration technique is best suited for multistage, pipelined converters with nominally identical stages. It corrects the most common causes of converter nonlinearity: errors on comparator trip-points, incorrect DAC levels and incorrect amplifier gains. In this paper, a mathematical description of the operation of such pipeline is first given. Terminology is defined and related to classic, uncorrected converters. Redundancy is described in a general way, and it is shown how it can correct comparator errors. Then, based on the formulas derived earlier, a complete digital correction scheme is introduced. An iterative algorithm called "accuracy bootstrapping" provides the required identification of the analog components, using much of the hardware already present in the system. Performance of the method is discussed, based on the (small) residual nonlinearity. An intuitive explanation of the convergence is given. Finally, fundamental limitations are discussed.

II. Generalized Mathematical Description

Fig. 1 shows a schematic of one converter stage. The incoming signal, V^{in}, is compared against a number of reference levels, V^{ref}, using a flash converter, which consists of a comparator string and a voltage divider that operates off the main reference. The comparator outputs provide a rough digital representation of the input voltage, in thermometer format. It will be called the local code, c. A flash section with M comparators can generate $M+1$ possible codes. The flash section is followed by a reconstructing digital/analog converter (DAC). Depending on the local code, one out of $M+1$ possible voltages, V^{DAC}, is subtracted from the input signal.

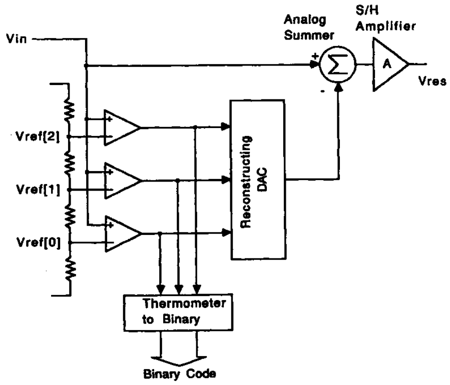


Fig. 1. Schematic of one converter stage.

The difference signal ($V^{\text{in}} - V^{\text{DAC}}$) is amplified by a sample/hold (S/H) amplifier. The amplifier gain, A, restores the difference signal to a level compatible with the input range of the next stage. The resulting signal will be called the residue, V^{res}. A can have any value greater than 1, positive or negative. A possible input-referred offset of the amplifier cannot be distinguished from a variation on the V^{DAC}'s and will not be discussed separately. The S/H action allows several stages to be cascaded in a pipelined fashion. Cascading more stages allows a higher resolution.

We will now consider a converter with L nominally identical stages, numbered 0 (last, or least significant stage) to $L-1$ (input, or most significant stage). For each stage (designated by its number, l, in subscript), the local code, c_l, can be written as a function of the input voltage V_l^{in}. We will designate the $M+1$ possible digital words for c_l by the integers 0 (for 00 $\cdots$ 0), 1 (for 10 $\cdots$ 0) through M (for 11 $\cdots$ 1).

$$c_l = \begin{cases} 0 & \text{for} \quad V_l^{\text{in}} < V_l^{\text{ref}}[0] \\ 1 & \text{for} \quad V_l^{\text{ref}}[0] \le V_l^{\text{in}} < V_l^{\text{ref}}[1] \\ \cdots & \\ M & \text{for} \quad V_l^{\text{ref}}[M-1] \le V_l^{\text{in}}. \end{cases} \tag{1}$$

For any given c_l, a general equation can be written for the residue V_l^{res} as a function of V_l^{in}.

$$V_l^{\text{res}} = (V_l^{\text{in}} - V_l^{\text{DAC}}[c_l])A_l. \tag{2}$$

Manuscript received October 25, 1993; revised April 24, 1994. This work was supported by Texas A&M University and Texas Instruments Inc. This paper was recommended by Associate Editor L. R. Carley.

E. G. Soenen is with the Mixed Signal Design Center, Texas Instruments Inc., Dallas, TX 75265 USA.

R. L. Geiger is with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA.

IEEE Log Number 9407653.

Or rearranged:

$$V_l^{\mathrm{in}} = V_l^{\mathrm{DAC}}[c_l] + \frac{V_l^{\mathrm{res}}}{A_l}. \tag{3}$$

From the structure of the pipeline, it is clear that

$$V_{l-1}^{\mathrm{in}} = V_l^{\mathrm{res}}. \tag{4}$$

The expression for the input voltage of stage l can then be expanded to

$$V_l^{\mathrm{in}} = V_l^{\mathrm{DAC}}[c_l] + \frac{V_{l-1}^{\mathrm{DAC}}[c_l - 1]}{A_l} + \frac{V_{l-1}^{\mathrm{res}}}{A_l A_{l-1}}. \tag{5}$$

By further expanding the same expression and evaluating it for $l = L-1$, the input voltage V^{in} of the pipelined converter with L stages can be written as (omitting the c_l for clarity):

$$\begin{aligned} V^{\mathrm{in}} &= V_{L-1}^{\mathrm{DAC}} + \frac{V_{L-2}^{\mathrm{DAC}}}{A_{L-1}} + \frac{V_{L-3}^{\mathrm{DAC}}}{A_{L-1}A_{L-2}} + \cdots \\ &+ \frac{V_0^{\mathrm{DAC}}}{A_{L-1}\cdots A_1} + \frac{V_0^{\mathrm{res}}}{A_{L-1}\cdots A_0}. \end{aligned} \tag{6}$$

This general equation is based on the assumptions that

1) All the parameters of the stage are fixed, i.e., time and signal invariant.
2) The gain of all interstage amplifiers is linear and free of hysteresis.
3) The difference operation occurs in a linear, signal-independent way.
4) The stage does not add any noise to the signal.

In a practical multistage A/D converter, the local codes of all the stages are combined to form the digital conversion result (CR), according to (6). The first terms of the equation are the useful ones. The last term is always neglected. It expresses the inherent quantization error, which originates from the fact that the analog residue from the last stage is not converted any further. It can be seen to decrease drastically as the number of converter stages (L) or the gain of the interstage amplifiers (A_l) are increased.

The same discussion applies to cyclic converters, in which the residue from a single stage is fed back to that same stage for successive conversions. Algorithmically, there is no difference between such arrangement and a pipeline. Equation (6) equally applies, as long as the stage subscript l is interpreted as designating a sample in time rather than a stage in a spatial arrangement. A peculiarity of the cyclic approach is that all parameters $V^{\mathrm{ref}}, V^{\mathrm{DAC}}$ and A are equal for successive stages. The calibration method to be discussed obviously works in cyclic structures as well.

III. Minimal and Base-2 Converters

CR is most easily obtained from the c_l when the stages are designed for what we will call "minimal" (as opposed to "redundant") operation. The procedure is further simplified when the converter is based on a radix of 2, i.e., when the gain of all interstage amplifiers is *exactly* a power of 2. Many multistage converter designs are of the minimal base-2 type.

A *minimal design* is characterized by an integer nominal gain value A and reference levels V^{ref} chosen so as to divide the input range (assumed to be between 0 and R) in as many equal parts as the absolute value of the gain, $|A|$. Hence, the number of comparators, M, is $|A| - 1$. The V^{ref} are integer multiples of a fixed voltage increment, $\Delta V = R/|A|$, so that $V_l^{\mathrm{ref}}[c] = (c+1)\Delta V = (c+1)R/|A|$, for $0 \le c < M$.

The number of DAC levels V^{DAC} is equal to $|A|$. The V^{DAC}'s of each stage are also integer multiples of $\Delta V = R/|A|$. For positive A the V^{DAC} are given by $V_l^{\mathrm{DAC}}[c] = c\Delta V = cR/|A|$. For negative A, this becomes $V_l^{\mathrm{DAC}}[c] = (c+1)\Delta V = (c+1)R/|A|$. In both cases, $0 \le c \le M = A-1$.

Fig. 2 shows the minimal designs for stages with nominal gains of 2, −2, −3 and 4. The figure also shows the transfer function of each stage, i.e., the relationship between V_l^{in} and V_l^{res}. Assuming ideal components, the transfer function of each stage exhibits a regular saw-tooth behavior. As long as the input signal to a stage is within the nominal input range, $0 \cdots R$, *its residue is guaranteed to be within the same range*, R. Since all stages are equal, the residue will fit the input range of the next stage. The last residue will be limited to $0 \cdots R$, and the resulting conversion error (further called quantization error), ε^{qu}, will be limited.

$$0 \le \varepsilon^{qu} \le \frac{R}{|A_{L-1}|\cdots|A_0|} \approx \frac{R}{|A|^L} \tag{7}$$

ε^{qu} represents the *inherent* uncertainty on the conversion result. In a classic, ideal N-bit converter, the corresponding quantization error range would be $R/2^N$, which is also defined as 1 *lsb* (for "least significant bit"). By extension, we are defining 1 *lsb* as follows:

$$1lsb = \frac{R}{|A|^L}. \tag{8}$$

Similarly, the relationship between the input range, R, and the range of the quantization error (1 *lsb*) can be expressed as the effective number of bits, N_{eff}.

$$N_{\mathrm{eff}} = \log_2\left(\frac{R}{1lsb}\right) = \log_2\left(\frac{R}{R/|A|^L}\right) = L\log_2(|A|). \tag{9}$$

The effective number of bits per stage, n_{eff} is given by:

$$n_{\mathrm{eff}} = \log_2(|A|). \tag{10}$$

For a minimal *base-2* converter, A is a power, n, of 2, so that $A = 2^n$. (Note that $n_{\mathrm{eff}} = n$.) The binary conversion result is particularly easy to obtain. It is sufficient to convert the c_l of successive stages from M-bit thermometer code to n-bit straight binary, and combine the L individual n-bit words into an N-bit (with $N = nL$) conversion result through concatenation. Furthermore, $0 \le \varepsilon^{qu} \le R/2^{nL} = R/2^N = 1lsb$. This expression is consistent with the conventional definition of an N-bit converter.

IV. Redundancy

It is difficult to control the reference levels V^{ref} of the flash sections precisely. Unfortunately, any effective trip-point variations will cause nonlinearities in the overall transfer curve

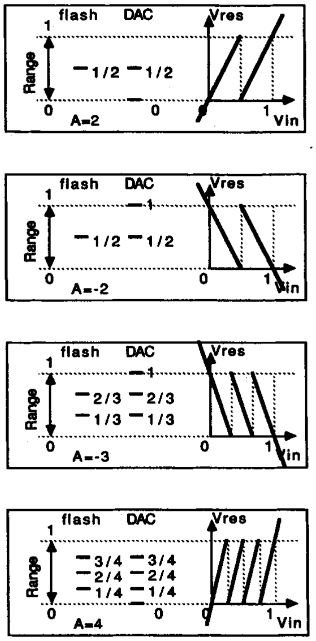


Fig. 2. Minimal designs.

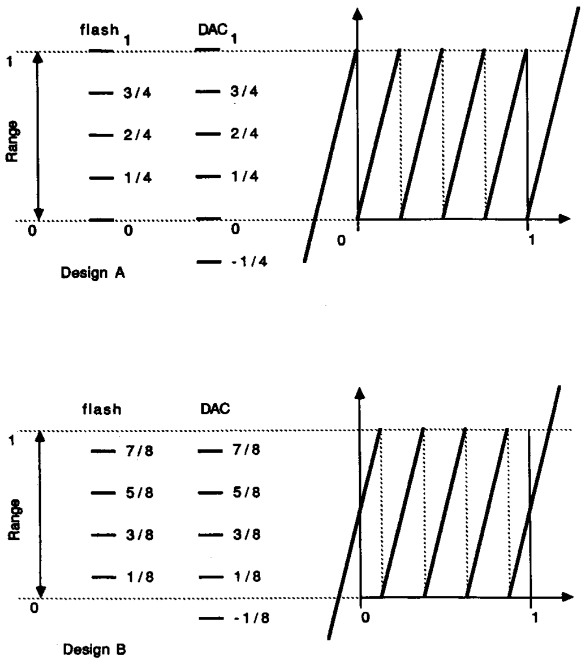


Fig. 3. Redundant stages.

of a minimal converter. Oddly enough, the errors are not caused directly by the incorrect reference levels, since the $V^{\rm ref}$'s do not appear in (6). Instead, reference level errors have an indirect effect, due to the fact that the residue of one stage exceeds the input range of the next stage.

Assuming ideal interstage amplifiers (linear over a wide range of input signals), (6) remains valid even when a comparator makes a wrong decision. But when a residue exceeds the input range of a subsequent stage, this causes the next residue to be even further out of range. Finally, the last residue gets so large that (7) becomes invalid. The quantization error becomes significant. In practice, some of the amplifiers may also clip, which makes matters even worse.

The problem can be avoided by increasing the input range of each stage, beyond the nominal output range of the previous stage. This guarantees that the residues would remain limited, as well as the overall quantization error (within 1 *lsb*). The input range can be increased using a design where $M > |A|-1$. Either the number of comparators is increased, or the interstage gain is decreased with respect to the minimal design. The first option is often preferred, because the nominal gain can remain integer, and even a power of 2. Two possible designs of redundant stages with nominal gain of 4 are shown in Fig. 3, as well as their transfer curves.

Design A uses two extra (redundant) comparators compared to the minimal case, at the top and bottom of the range [1]. $V_l^{\rm ref}[c] = (c+1)\Delta V = (c+1)R/|A|$, for $-1 \le c < M+1$ and $V_l^{\rm DAC}[c] = c\Delta V = cR/|A|$, for $-1 \le c \le M+1$. One can verify that this provides an overrange capability of $\pm R/|A|$. We will call this arrangement "**minimal + 2**."

Design B uses one redundant comparator compared to the minimal case and offsets both the $V^{\rm ref}$ and the $V^{\rm DAC}$ by one half ΔV. $V_l^{\rm ref}[c] = (c+1/2)\Delta V = (c+1/2)R/|A|$, for $0 \le c < M+1$ and $V_l^{\rm DAC}[c] = (c-1/2)\Delta V = (c-1/2)R/|A|$, for $0 \le c \le M+1$. In literature, this arrangement has been described as implementing a "Redundant Signed Digit" (RSD) [2], [3] algorithm. We will call it "**minimal + 1**." One can verify that this provides an overrange capability of $\pm R/(2|A|)$.

Either redundant scheme requires some modification of the method used to derive the conversion result from the c_l for base-2 converters, resulting in slightly more complex logic [1]. However, the gain in robustness can be spectacular.

V. Generalized Digital Error Correction

Redundancy solves the problem of variability in $V^{\rm ref}$'s, but **not** in the $V^{\rm DAC}$'s and the A's. However, based on (6), a digital error correcting scheme that compensates for DAC and gain errors can be derived. Equation (6) is general. It will provide a correct approximation for the input voltage, as long as its terms are computed using the *actual* values of all converter components, rather than the nominal ones.

A more powerful approach consists of associating small digital look-up tables with each stage of the converter, like depicted in Fig. 4. Each table is addressed by the local code of the stage, and generates one term of (6). The terms are summed in a pipelined fashion in order to form the conversion result. No matter what the *actual* values of gains and DAC levels are, the converter can be linearized (within the limits of the quantization error) by using appropriate digital data ("coefficients") in the look-up tables. No tuning of analog components is necessary, as long as the coefficients are computed accurately.

To fit the model of Fig. 1 and keep the quantization error within bounds despite the presence of component errors, enough redundancy must be built into each stage. For a "minimal + 1" pipeline, correct operation is guaranteed when

$$(\varepsilon^{\rm DAC}+\varepsilon^{\rm DAC})RA+\varepsilon^{A}R<\frac{R}{2A}. \tag{11}$$

Where R is the nominal input range, $\varepsilon^{\rm ADC}$ is the worst-case flash error, $\varepsilon^{\rm DAC}$ is the DAC error (both relative to R) and ε^{A} is the worst-case relative gain error. It should be noted that the tolerance against flash and DAC errors decreases with the *square* of the interstage gain. Hence, pipelines with an interstage gain of only 2, or one "bit" per stage, are particularly robust.

The coefficients in the arrangement of Fig. 4 are nominally different for each stage. (Successive terms of (6) decrease in absolute value.) This is a draw-back: since the numeric range of each stage is different, the digital hardware cannot simply be duplicated from stage to stage. This complicates the design, and the use of the self-calibration algorithm to be described below. To improve the structure, (6) is rewritten as (A is the *nominal* interstage gain):

$$\begin{aligned}V^{\rm in}=&V_{L-1}^{\rm DAC}1\frac{A^{(L)}}{A^{(L)}}+V_{L-2}^{\rm DAC}\frac{A}{A_{L-1}}\frac{A^{(L-1)}}{A^{(L)}}\\&+V_{L-3}^{\rm DAC}\frac{A^{(2)}}{A_{L-1}A_{L-2}}\frac{A^{(L-2)}}{A^{(L)}}\\&+\cdots V_{0}^{\rm DAC}\frac{A^{(L-1)}}{A_{L-1}\cdots A_{1}}\frac{A}{A^{(L)}}\\&+\frac{V_0^{\rm res}}{A_{L-1}\cdots A_0}\end{aligned} \tag{12}$$

$$\begin{aligned}V^{\rm in}=&\Bigg[\Bigg[\Bigg[\Bigg[\Bigg[\frac{V_{L-1}^{\rm DAC}}{A^{(L)}}1\Bigg]A+\frac{V_{L-2}^{\rm DAC}}{A^{(L)}}\frac{A}{A_{L-1}}\Bigg]A\\&+\frac{V_{L-3}^{\rm DAC}}{A^{(L)}}\frac{A^{(2)}}{A_{L-1}A_{L-2}}\Bigg]A\\&+\cdots\Bigg]A+\frac{V_{0}^{\rm DAC}}{A^{(L)}}\frac{A^{(L-1)}}{A_{L-1}\cdots A_{1}}\Bigg]A\\&+\frac{V_0^{\rm res}}{A_{L-1}\cdots A_0}.\end{aligned} \tag{13}$$

This equation expresses the conversion result in a recursive way. It will be shown that, when translated into hardware, this form results in identical logic for each stage. We will redefine the table coefficients, and further call them "weights," W. The "ideal" weights, W^i, (calculated based on perfect knowledge of the actual component values) are defined as:

$$W_l^i[c]=V_l^{\rm DAC}[c]\frac{A^{(L-1-l)}}{A_{L-1}\cdots A_{l+1}} \tag{14}$$

The conversion result can now be rewritten as follows (some superscripts and indices are dropped for notational convenience):

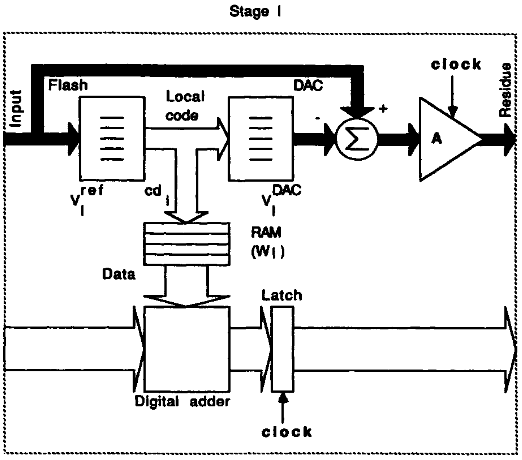


Fig. 4. Converter stage with look-up table.

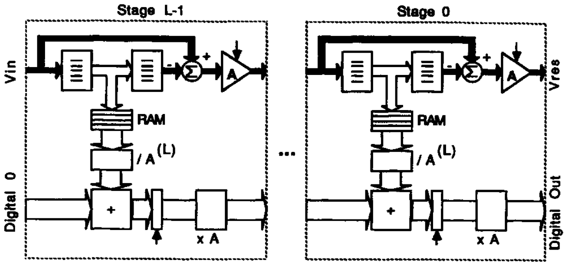


Fig. 5. Nominally identical look-up tables.

$$\begin{aligned}CR=&\Bigg[\Bigg[\Bigg[\Bigg[\Bigg[\frac{W_{L-1}}{A^{(L)}}\Bigg]A+\frac{W_{L-1}}{A^{(L)}}\Bigg]A\\&+\ \frac{W_{L-3}}{A^{(L)}}\Bigg]A+\cdots\Bigg]A+\frac{W_0}{A^{(L)}}\Bigg]A\end{aligned} \tag{15}$$

It is clear that the weights associated with each stage are nominally identical, and nominally equal to the $V^{\rm DAC}$ values, since the correction factors A/A_{L-1}, $A^{(2)}/(A_{L-1}A_{L-2})$ etc., are nominally equal to unity. In practice, component variations are small and the values of the weights will also remain close to nominal. The architecture that calculates of the conversion result according to (15), is shown in Fig. 5. The look-up tables contain the W values. The logic of each stage is identical. Draw-backs of this scheme versus the one of Fig. 4, are the extra divisions by $A^{(L)}$ and multiplications by A. However, if A is chosen as a power of 2, these operations can be performed by a mere shifting of bit lines, without additional computational hardware.

It should be noted that the weights computed according to (6) or (14) are not necessarily fractional values. Yet in a practical circuit, they must be represented by a limited number of bits. In addition, the summation of (15) will be performed to a limited precision. This means that a certain evaluation error is going to be made on the conversion result, due to truncation. This truncation error is statistically independent of the inherent quantization error of the converter. Its magnitude can be kept well below the quantization error by proper dimensioning of internal digital registers and buses. However, a significant truncation error ($\pm 1/2lsb$) is often made at the output of the converter. System-level considerations often limit the number of converter output lines to its nominal number of bits, $N_{\rm eff}$, rather than to its full internal numerical resolution.

VI. The "Accuracy Bootstrapping" Algorithm

Digital correction by local look-up tables based on (15) requires knowledge of the weights W. They could be calculated analytically according to (14), but this requires accurate knowledge of the *actual* gains and DAC levels of the system. It is conceivable to individually measure these parameters using precision external equipment. Although feasible for the one-time calibration of simple converters at the factory, the method is intensive and does not allow for periodic in-circuit recalibration. These limitations are removed by the "**accuracy bootstrapping**" [4] algorithm. It simultaneously measures the converter components and calculates the required weights in an iterative fashion (hence the name given to the method). Its power is impressive, because it uses the data path already present to calculate the conversion result. *The basic idea of the algorithm is to individually measure all the DAC levels of each converter stage, using the remaining stages of the pipeline. The measurements are used to update the look-up tables of that stage, and the process is repeated until each stage has been calibrated.*

At a first glance, it may seem unlikely that any accuracy could be gained this way. In practice, it has been observed that for specific configurations, the procedure results in an iterative numerical problem that is extremely stable and converges to the desired result very fast. The algorithm requires rearranging of converter stages into a circular structure. The analog residue of the last stage (stage 0) is fed back to the first stage ($L-1$), and so is the digital bus carrying the conversion result. The arrangement uses the nominally identical stages of Fig. 5. It is now possible to inject an analog input signal at the input of *any* stage, and use the circular sequence of stages back to that same point to perform the analog to digital conversion.

Within each stage, the possibility is added to by-pass the flash comparators and control the reconstructing DAC externally rather than through the local codes c. It is also possible to replace the analog input signal by a fixed potential, $V^{\rm fix}$. The algorithm requires the use of a DAC/subtractor combination like shown conceptually on Fig. 6. The DAC is composed of a fixed voltage, used to bias the gain stage, and a number of voltage increments, which are selectively enabled and subtracted from the input voltage. One can verify that the value of each increment corresponds to one $\Delta V^{\rm DAC}$, nominally equal to $1/|A|$. $\Delta V_l^{\rm DAC}[c]$ is defined as the *differential* DAC level between the cth DAC level and the $(c-1)$th DAC level of stage l, or $\Delta V_l^{\rm DAC}[c] = V_l^{\rm DAC}[c] - V_l^{\rm DAC}[c-1]$. The more comparators turn on (the larger the signal), the more increments are subtracted. The depicted stage is of type "minimal + 1", with an input range between 0 and 1, and a gain of 2. The two voltage increments of 0.5 each result in differential DAC levels, $\Delta V_l^{\rm DAC}[1]$ and $\Delta V_l^{\rm DAC}[2]$, both nominally equal to 0.5. The resulting nominal $V^{\rm DAC}$ values are -0.25, 0.25 and 0.75.

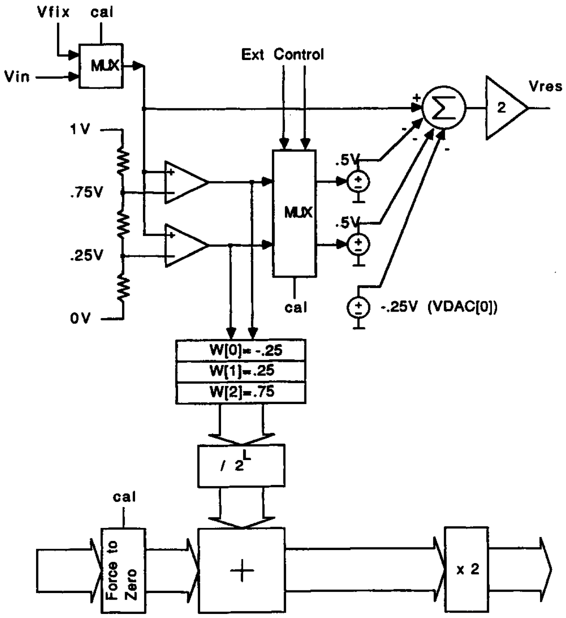


Fig. 6. Stage, modified for calibration.

The accuracy bootstrapping algorithm estimates weight values W^e (superscript e stands for "estimate") iteratively. The procedure proceeds as follows:

1) The estimated weights of all stages, W^e, are initialized to their nominal values: $W_l^e[c] = V_l^{\rm DAC,nom}[c]$.
2) The last stage (stage 0) is calibrated first. The analog input of that stage is held at the fixed potential, $V^{\rm fix}$, while *none* of the voltage increments are enabled. The remaining $L-1$ stages of the circular structure $(L-1, L-2, \cdots 1)$, as well as the flash section of stage 0, are used to determine the conversion result, $C_0[0]$. This zero-level measurement will be used to cancel any systematic offset in the stage, as well as the unpredictability of $V^{\rm fix}$.
3) The analog input of the last stage is held at $V^{\rm fix}$ while only the *first* voltage increment is enabled. The same $L-1$ stages of the circular structure $(L-1, L-2, \cdots 1$, and flash section of 0) are used to determine the new conversion result, $C_0[1]$. The *second* voltage increment is then enabled by itself and the same procedure is followed to determine $C_0[2]$. In general, the procedure is repeated until the M increments have been measured $(C_0[1] \cdots C_0[M])$. It is clear that the fixed input potential must be chosen such that the input signals under 2,a) and 2,b), do not over-range the gain stage. Nominally, $V^{\rm fix} = 1/|A|$ (for $A < 0$ and for $A > 0$). Each $C_0[c]$ value represents a measurement of $A_0 \Delta V^{\rm DAC}[c]$, since each voltage increment was multiplied by the gain of stage 0 before being converted by the sequence of stages starting from stage $L-1$.

4) The actual values of the voltage increments of stage 0 can now be estimated, based on the measurements. The value $D_0[1] = (C_0[1] - C_0[0])/A$ is used as an estimate for $\Delta V_0^{\rm DAC}[1]$. (Subtracting $C_0[0]$ cancels the effect of offset or incorrect $V^{\rm fix}$.) Similarly, $D_0[2] = (C_0[2] - C_0[0])/A$ is an estimate for $\Delta V_0^{\rm DAC}[2]$. In general, $\Delta V_0^{\rm DAC}[c]$ is estimated as $D_0[c] = (C_0[c] - C_0[0])/A$, for $c = 1 \cdots M$. It should be noted that the D_c's are computed by dividing differential measurements by the *nominal* value of the gain, A. As a result of using this rather than the (unknown) *actual* gain value, A_0, the $\Delta V_0^{\rm DAC}$'s are systematically over-estimated by A_0/A!
5) The D_0 values (estimated $\Delta V_0^{\rm DAC}$'s) are now used to estimate the $V_0^{\rm DAC}$'s. The estimates will form the new weights for stage 0 $(W_0^e[c])$. $W_0^e[0]$ is arbitrarily kept equal to its nominal value. The other DAC levels are computed by adding the previously obtained $D_0[c]$. $W_0^e[1] = W_0^e[0] + D_0[1], W_0^e[2] = W_0^e[1] + D_0[2]$, or in general: $W_0^e[c] = W_0^e[c-1] + D_0[c]$, for $c = 1 \cdots M$. All original (nominal) $W_0[c]$ values in the look-up table of stage 0 are now replaced by the newly determined ones. This concludes the calibration of stage 0.
6) Next, the procedure used on stage 0 is repeated to calibrate stage 1. The voltage increments of stage 1 are measured, using the converter formed by stages 0, $L-1, L-2, \cdots 2$, and the comparator section of stage 1. The measured values $C_1[c]$ are used to estimate the differential DAC levels: $D_1[c] = (C_1[c] - C_1[0])/A$. Finally, the new weights $W_1[c]$ are computed and used to replace the original weights of stage 1.
7) The same procedure is repeated to calibrate the stages $l = 2 \cdots L-1$. The voltage increments of stage l are measured, using the converter formed by stages $l-1, l-2, \cdots 0, L-1 \cdots l$. The previously calibrated stage is always the first (most significant) stage in the converter used to calibrate the next one. Whenever the weights of stage $L-1$ have been updated, the first step of the calibration is complete.
8) After one iterative calibration cycle through all the stages, one could repeat the procedure starting from step 2), to further refine the estimates of the weights. However, simulations indicated that for practical values of the analog errors (within 1%), the digitally compensated pipelined converter will exhibit a very linear transfer characteristic after one cycle. More cycles do not usually yield significant improvements.
9) Since no external reference standard was used during the calibration, the resulting transfer characteristic (although linear) may be subject to a residual offset or gain error. It will be shown below how this can be eliminated as well.

Correct calibration can still be obtained when the analog components (in particular the interstage amplifiers) of the pipeline are noisy. The influence of the noise upon the calculation of the weights can be reduced to an arbitrarily low level by averaging a number of successive measurements for each $D_l[c]$ value. Of course, the conversion result of the pipeline will be subject to the noise during subsequent continuous operation, but any systematic (harmonic) distortion will be limited.

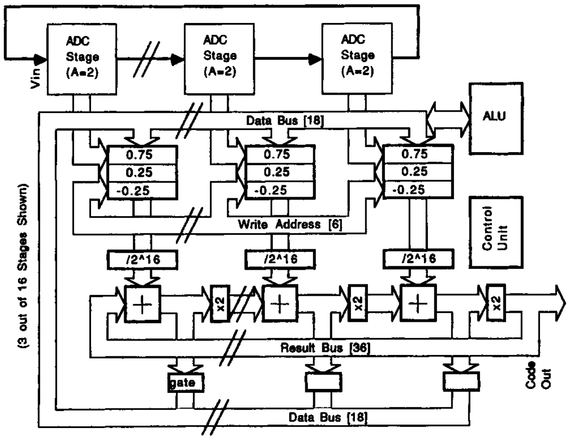


Fig. 7. Calibration hardware.

Fig. 7 shows a possible block diagram of a monolithic 16 b pipelined converter, including look-up tables and pipelined digital data path for real-time operation. The addition of a controller and a simple arithmetic unit allows on-chip implementation of the calibration cycle. Obviously, correction and calibration can also be performed off-chip, with dedicated logic or a general-purpose computer. The last approach would make it possible to integrate a relatively small, analog data path as a peripheral for a complex digital system like a microcontroller.

VII. Scaling and Linearity Criterion

The residual gain and offset error that remain after step 5) of the accuracy bootstrapping procedure can be eliminated through the appropriate scaling of all weights. The procedure is based on the measurement of the two fixed reference voltages (zero and full-scale). One can easily verify the following two scaling rules:

1) Adding a fixed value, k, to (or subtracting from) all weights associated with *one* stage l, results in an offset (vertical shifting) of the transfer curve, by $k/A^{(L-1-l)}$. (The fact that the overall linearity is not affected, is the reason why in step 2.d) of the algorithm, $W_l^e[0]$ could be arbitrarily set to its nominal value.)
2) When all weights associated to *all* stages are multiplied by a fixed value, s, the slope (gain) of the transfer curve is multiplied by s. (A change in offset may result.)

The offset and gain error can be corrected using the conversion result for the nominal zero level (M_0), and for the nominal full-scale input of the converter (M_R). M_0 is the overall offset before scaling. $[M_R - M_0]/R$ is the gain error.

For the theoretical evaluation of the algorithm, one needs a criterion that isolates residual linearity errors due to incorrect computation of the weights. These are *calibration* errors, different from the other kinds of conversion errors. Sweeping an input signal over the whole converter range and looking for the worst-case difference between output value and known input, would lump the calibration errors (1) with the other errors: the (inherent) *quantization* error (2), the residual *gain* (3) and *offset* (4) errors and the *truncation* error (5), due to possible truncation of the conversion result to a limited number of bits.

Traditional linearity criteria could be used [5], [3]. Differential or integral nonlinearity (DNL or INL) are determined through a code density test and are insensitive to quantization, gain and offset errors. However, DNL and INL would need to be slightly redefined for this class of converters, which has more than 2^N transition levels (due to the redundancy) and for which small, local nonmonotonicities may exist.

A generalized definition of INL could be: the amount by which the range of the total approximation error *exceeds* 1 *lsb* (inherent quantization error). Similarly, DNL can be defined as: the amount by which the maximum difference between two consecutive digital outputs *exceeds* 1 *lsb.*

Signal-to-noise or signal-to-distortion ratios could also be calculated, like is common for sigma-delta converters (using a sinusoidal input signal and a discrete Fourier transform). The distortion would reflect the residual nonlinearity, while the noise would lump quantization and truncation errors. However, both criteria can be impractical in simulations, due to the need to perform many conversions.

The criterion we will use instead, determines a *worst-case* calibration error, based directly on a comparison between the *estimated* and the *ideal* weights. The estimated weights are computed through accuracy bootstrapping; the ideal weights are the theoretical weights, calculated according to (6) or (14), based on the exact knowledge of the *actual* parameters of the system. (Ideal weights should not be confused with nominal weights, which are based on nominal, error-free component values.) In the process, gain and offset errors are obtained as well.

The philosophy behind the criterion is that the "best" (most linear) approximation of the input signal is obtained with a converter using *ideal* weights. (In a statistical sense, the mean square approximation error in (6) is indeed minimized, since it can be assumed that the probability distribution of the quantization error is uniform for a uniform distribution of input signals.) Any weight different from the ideal value, contributes to the overall linearity error. For the calculation of the overall error, the estimated weights are first scaled (to exclude a possible gain error), and then compared with the ideal weights on a differential basis (to exclude possible offsets).

1) The $L \times (M+1)$ array of *ideal* terms $T_l^i[c]$ is calculated based on (6):

$$T_l^i[c] = \frac{V_l^{\mathrm{DAC}}[c]}{A_{L-1} \cdots A_{l+1}} = \frac{W_l^i[c]}{A^{(L-1-l)}}. \tag{16}$$

2) The $L \times (M+1)$ array of *estimated* terms $T_l^e[i]$ is calculated based on the estimated weights, $W_l^e[i]$ (as defined in the discussion about the accuracy bootstrapping algorithm). This requires division by as many times the nominal gain as there are preceding stages.

$$T_l^e[c] = \frac{W_l^e[c]}{A^{(L-1-l)}}. \tag{17}$$

3) The maximum and minimum value that can possibly be represented at the converter output, using the *ideal* weights, are determined. The maximum value, $C_{\max}^i$, is the sum of the largest possible term of each stage. The minimum value, $C_{\min}^i$, is the sum of the smallest possible term of each stage. (These values are not the same as top and bottom of the input range.)

$$\begin{aligned} C_{\max}^i &= \sum_{l=0}^{L-1} \mathrm{Max}_{c=0}^{M}\,(T_l^i[c]), \\ C_{\min}^i &= \sum_{l=0}^{L-1} \mathrm{Min}_{c=0}^{M}\,(T_l^i[c]). \end{aligned} \tag{18}$$

4) Similarly, the maximum and minimum value that can possibly be represented at the converter output, using the *estimated* weights, are determined:

$$\begin{aligned} C_{\max}^e &= \sum_{l=0}^{L-1} \mathrm{Max}_{c=0}^{M}\,(T_l^e[c]), \\ C_{\min}^e &= \sum_{l=0}^{L-1} \mathrm{Min}_{c=0}^{M}\,(T_l^e[c]). \end{aligned} \tag{19}$$

5) The overall relative gain error, ε^A, of the converter is estimated. This estimate may be slightly pessimistic when compared with a least-squares fit approach, but is much easier to calculate.

$$\varepsilon^A = \frac{[C_{\max}^e - C_{\min}^e]}{[C_{\max}^i - C_{\min}^i]}. \tag{20}$$

6) The estimated terms are scaled, to compensate them for the effect of the gain error:

$$T_l^s[c] = T_l^e[c]\varepsilon^A, \quad \text{for}\, l = 0 \cdots -1 \quad \text{and} \quad c = 0 \cdots M. \tag{21}$$

7) The worst-case range of calibration errors is determined for each of the L terms of (6), through the comparison between estimated and ideal values. The scaling performed above removes the effect of a possible gain error. A possible offset, common to all possible values of one term, is taken out of the picture by comparing the two sets of values differentially. *A linearity error occurs when one or more flash sections of the converter flip from one output code to the next, but the corresponding digital terms do not correctly reflect the change in subtracted DAC voltages.* For a specific stage, the range of this error is the maximum difference in "step" between any two estimated terms, and the corresponding "step" in ideal terms. Summation of these errors over all stages yields the overall worst-case calibration error.

$$\varepsilon^{cal} = \sum_{l=0}^{L-1} \mathrm{Max}_{k,j=0}^{M}\,((T_l^s[k] - T_l^s[j]) - (T_l^i[k] - T_l^i[j])). \tag{22}$$

This calibration error can be expressed in lsb units through division by $R/|A|^L$ (1 lsb, according to (8)). Alternatively, the number of effective bits can be calculated by taking the base-2 logarithm of the ratio $R/\varepsilon^{\text{cal}}$. Since in practice not each error of each term reach their maximum at the same time, the criterion described here is pessimistic. A code density test (INL measurement) will reveal slightly smaller nonlinearity values. We used the stricter criterion in all simulations presented below.

8) If desired, the overall offset error, ε^{off}, at the mid-point of the converter transfer curve, can be estimated:

$$\varepsilon^{off} = \frac{[C^a_{\max} - C^i_{\max}] + [C^a_{\min} - C^i_{\min}]}{2}. \tag{23}$$

VIII. Simulation Results

The following simulations demonstrate the ability of the accuracy bootstrapping algorithm to accurately calibrate a variety of pipelined converter architectures. Since the algorithm is iterative in nature (successive converter stages are calibrated using an imperfectly calibrated sequence), it is imperative to determine under what conditions the procedure will converge.

High-level Monte-Carlo simulations proved to be a valuable statistical tool for this analysis. Many converter configurations with random or systematic error patterns were simulated. For a particular batch (a certain architecture), all errors were randomly modified before each run, and the resulting nonlinearity figures (in lsb) after a complete calibration cycle were tallied. It was observed that the distribution of residual calibration errors was always very close to normal. This was to be expected, since the final figure is the result of a complex interaction between many independent error variables. The mean and standard deviation of the residual error can be used as a design or reliability criterion for particular architectures. For example, if a Monte-Carlo simulation reveals that a particular architecture with a particular set of worst-case errors has a mean residual linearity error after calibration of 0.7 lsb and a standard deviation of 0.2 lsb, the architecture would be adequate for 2 lsb worst-case INL in the six-sigma sense $(0.7 + 6 * 0.2 < 2)$.

Some results are summarized in Table I. The simulations can be replicated using any high-level programming language. All the results discussed here were obtained for "minimal+1" architectures, on which accuracy bootstrapping was found to perform most reliably. The table lists the number of stages (L) and the nominal gain (A). These two parameters set all other nominal parameters (flash reference and DAC levels) of the converter for a particular batch of Monte-Carlo simulations. The statistical component variations were simulated by adding random errors to the nominal parameters before each run, using a random number generator. The distribution of errors is binary (i.e., either maximally positive or maximally negative). The table lists the magnitude of the error on all gains (eA, in %), reference levels and DAC levels (eV and eD, in % of the input range). Input-referred white noise introduced in the stages was modeled by adding an error to each signal, again according to a binary distribution. The magnitude of the white noise is expressed in lsb.

TABLE I
Simulation Results

L	A	eA	eV	eD	noise	Nav	Nit	bits	mean	sigma
16	2	1	1	1	.1	1	1	16	.68	.22
16	-2	1	1	1	.1	1	1	16	.69	.21
16	-2	1	1	1	0	1	1	16	.67	.20
16	2	1	1	1	.1	1	2	16	.55	.20
16	-2	1	1	1	.1	1	2	16	.56	.20
16	-2	1	1	1	2	1	1	16	2.32	.71
16	-2	1	1	1	2	8	1	16	.91	.26
16	-2	3	3	3	.1	1	1	16	1.33	.57
16	-2	3	3	3	.1	1	2	16	.59	.22
4	4	1	1	1	.1	1	1	8	.30	.09
8	4	1	1	1	.1	1	1	16	.39	.12
20	-2	1	1	1	.1	1	1	20	.79	.24
24	2	1	1	1	.1	1	1	24	.97	.29
10	3	1	1	1	.1	1	1	15.85	.46	.14
16	2	.5	.5	.5	.1	1	1	16	.59	.19
24	-4	1	1	1	.1	1	1	48	2.53	1.26
24	-4	1	1	1	.1	1	2	48	.29	.10

The variables characterizing the accuracy bootstrapping algorithm itself are the number of averages used for the measurement of each differential DAC level (Nav) and the number of iterations (Nit), which are also listed in the table. For easy reference, the nominal number of bits of each architecture ($bits$) are included in the table as well.

Each Monte-Carlo batch consisted of 1000 simulation runs, with each time a new, random combination of errors. The empirical statistics of each batch are summarized as the mean and standard deviation of the residual linearity error (in lsb), like defined in (22). These simulations show that the residual calibration errors in "minimal + 1" architectures, subject to "realistic" component errors (1%), are comparable to the inherent quantization error (1 lsb), at both the 16 b and the 24 b level. It should be noted that the use of a binary rather than a normal error distribution represents a pessimistic, worst-case approach. Calibration also appears to work on "minimal + 2" architectures, but results in larger residual errors and less reliability (larger sigma). A pipelined converter that seems to use a somewhat similar calibration technique with one comparator per stage and a gain smaller than 2, has recently been reported [6]. Few details are presently available.

The criterion ((22)) also allows calculation of the linearity *between* calibration steps. Fig. 8 shows the linearity for a number of 16 b (16 stages with gain of 2) "minimal + 1" converters, expressed in effective bits ($\log_2(R/\varepsilon^{cal})$). Each of them had initial relative errors of 1% on all components. The figure shows the linearity before calibration, then each time after a stage has been calibrated (the newly calibrated stage being first in the pipeline). This is repeated for two passes. It should be noted that at the beginning of the second pass, the linearity suddenly drops, before picking up again. The cause of the drop will become clear in the next section. It will be shown how the weights associated with each stage are calculated in such a way that when exactly one pass of the algorithm is performed, all weights are scaled by a same amount. This results in a linear overall transfer curve, despite the presence of gain errors on each interstage gain. When a second pass of the algorithm is started, a new gain mismatch is introduced, which is only resolved after that pass is completed.

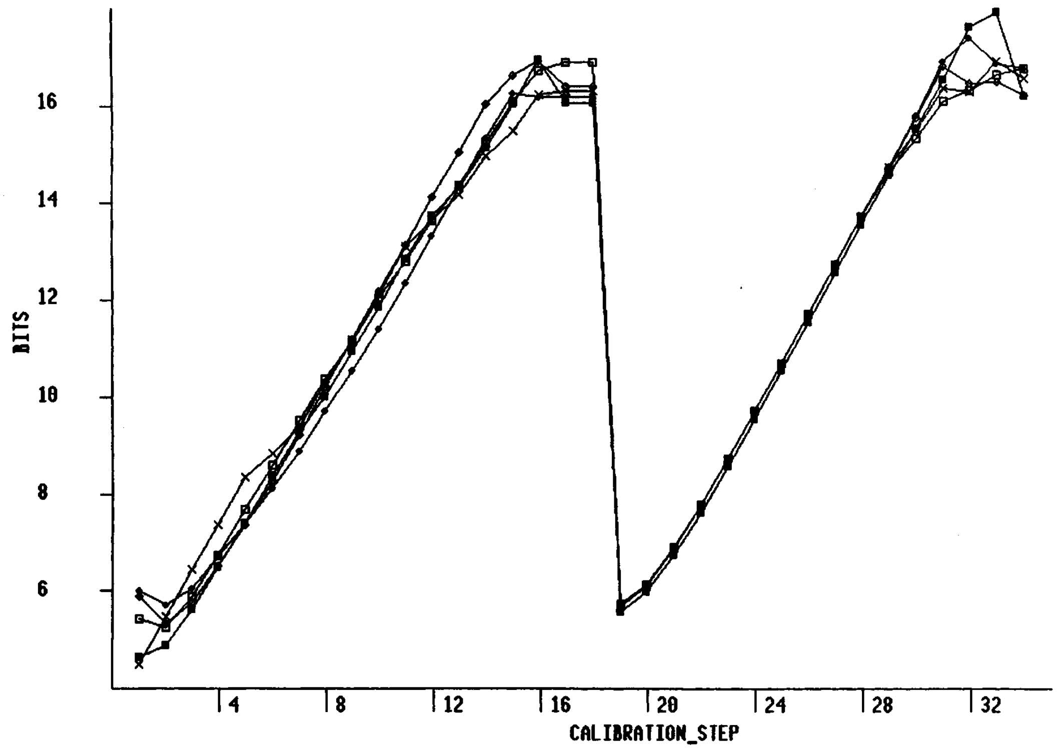


Fig. 8. Progression of the calibration.

IX. Intuitive Explanation of the Convergence

A closed, mathematical proof of convergence for the accuracy bootstrapping algorithm, is not straight-forward. In this section, intuitive considerations are discussed that help understand the mechanisms involved. They can best be explained separately for the different kinds of errors. Only gain and DAC errors need to be considered, and not flash errors, since the reference levels do not appear in (6) or (15). We will assume a "minimal + 1" converter. The compensation of gain errors can be explained as follows. When stage 0 is calibrated, its differential DAC levels $\Delta V_0^{\mathrm{DAC}}$ are estimated based on the measured values C_0 (step 2,c) of the algorithm):

$$D_0[c] = (C_0[c] - C_0[0])/A \tag{24}$$

Since the C_0 values were measured using the converter itself, they can be written in the format of (15), which in turn can be rewritten as:

$$CR = W_{L-1} + \frac{W_{L-2}}{A} + \frac{W_{L-3}}{A^{(2)}} + \cdots + \frac{W_0}{A^{(L-1)}} \tag{25}$$

Expanding $C_0[c]$ and $C_0[0]$ in this manner yields:

$$D_0[c] = \left(W_{L-1}[M] + \frac{W_{L-2}}{A} + \frac{W_{L-3}}{A^{(2)}} + \cdots \frac{W_0}{A^{(L-1)}}\right) \Big/ A - \left(W_{L-1}[0] + \frac{W_{L-2}}{A} + \frac{W_{L-3}}{A^{(2)}} + \cdots \frac{W_0}{A^{(L-1)}}\right) \Big/ A. \tag{26}$$

It can be verified that since $C_0[0]$ is measured with only V^{fix} applied (steps 2,a) and 2,b) of the algorithm), the input signal to the next stage $(L-1)$ is nominally equal to R (full-scale), which results in a comparator code $c = M$ for stage $L-1$. The measurements for which a differential DAC level is subtracted $(C_0[1] \cdots C_0[M])$ are characterized by a nominal input voltage of 0 to stage $L-1$, which results in a comparator code $c = 0$ for stage $L-1$. (The opposite is true if the gain is negative.) However, all the other terms are nominally common between $C_0[0]$ and $C_0[c]$, with $c = 1 \cdots M$. As a result, when the difference is taken in (26), most terms cancel out, except for the two W_{L-1} terms, and some lower-order terms, which we will neglect. (The smaller the initial errors, the better the approximation). This simplification yields:

$$D_0[c] \approx \frac{W_{L-1}[M] - W_{L-1}[0]}{A}. \tag{27}$$

This shows that the DAC levels (obtained by summing together a number of D-terms) of stage 0 are essentially estimated using the difference between the largest and the smallest weight of stage $L-1$, scaled by A. Since A is used instead of the (unknown) value of A_0, each D_0 value (and hence each

W_0) is overestimated by a factor A_0/A. Similarly, one can derive an expression for $D_1[c]$ (calibration of stage 1).

$$D_1[c] \approx \frac{W_0[M] - W_0[0]}{A}. \tag{28}$$

It is clear that because of the division by A, each D_1 value (and hence each W_1) is overestimated by a factor A_1/A, on top of the factor A_0/A already present in the estimates for W_0. As a result, the W_1 values are overestimated by $A_1A_0/A^{(2)}$. Similarly, after calibration, the W_2 values are overestimated by $A_2A_1A_0/A^{(3)}$, and so on. The W_{L-1} are overestimated by $A_{L-1}\cdots A_0/A^{(L)}$. Comparison with (14) shows that the ideal W_l's *should have* been obtained from the ideal $V^{\rm DAC}$'s through multiplication with $A^{(L-1)}/A_{L-1}\cdots A_1, A^{(L-2)}/A_{L-1}\cdots A_2$ etc. This means that after one calibration cycle through all stages, **all** W_l have effectively been overestimated by $A^{L-1}\cdots A_0/A^{(L)}$. Since it was shown that multiplication of all weights by a constant factor does not affect linearity, this illustrates that accuracy bootstrapping achieves linearization despite the presence of gain errors. It explains why a residual gain error is left after calibration. A similar reasoning can be applied to explain the discontinuity in linearity after one pass of the algorithm, as shown in Fig. 8.

The compensation of DAC errors can be considered independently from the gain errors, but can be explained in a similar way. The differential DAC levels of stage 0 are estimated using measurements (D_0) that are dominated by $(W_{L-1}[M] - W_{L-1}[0])/A$. Initially, nominal values will be used for the W_{L-1}'s. If the actual $V_{L-1}^{\rm DAC}$ values are off, so that $(W_{L-1}[M] - W_{L-1}[0]) = f_0(W_{L-1}^i[M] - W_{L-1}^i[0])$ (the ideal weights W^i are defined in (14)), the *range* of the W_0 will be overestimated by f_0, *as if* a linear scaling had taken place. Similarly, the range of the W_1's, based on a measurement dominated by W_0 terms, will be overestimated by f_0. Independently however, there may still be a measurement error, due to some lower-order terms we neglected. This can be expressed by an additional factor f_1, which expresses the relative measurement error on $(V_1^{\rm DAC}[M] - V_1^{\rm DAC}[0])$, *on top* of the error due to the W_0's.

This reasoning can be continued, and it will appear that each new set of weights will be scaled in a way that incorporates the measurement errors on all previously calculated weights, plus a new factor f to account for the current measurement error. The idea is that an error in the estimation of earlier W's is not critical, as long as later W's remain consistent. The "scaling" of stages towards the front of the pipeline must match the scaling of all stages behind them. The factors f_l tend to get closer and closer to unity for successive stages, because more stages are matched and the accuracy of the pipeline improves. The process eventually results in a linearization of the overall transfer function.

X. Noise and Nonlinearity

The proposed architecture, together with the accuracy bootstrapping algorithm, solves the problem of unpredictable flash transition levels, DAC levels, and interstage gains. These parameters are usually the hardest to control in high-resolution converter designs. A great advantage of this solution, is that the calibration is performed using the actual analog data path (analog pipeline of stages) of the converter. As a result, all parasitics are effectively taken into account: the calibration is performed at actual speed, with all switches, multiplexers etc. present.

As long as the analog stages fit the model of Fig. 1, it does not appear that there is a it fundamental limit to the final linearity that can be achieved through accuracy bootstrapping, *no matter how many stages are cascaded.* Simulations have shown successful calibration of 48 b converters (although impossible to realize in practice), with 1% initial error on all components. However, the model does not include effects like noise, interference, parasitic coupling from stage to stage (e.g., through the power supply), nonlinear gain, component drift etc. These must be kept under control through sound analog design. Present state-of-the-art CMOS technology allows the design of fully differential switched-capacitor amplifiers with linearity and noise performance to the 14 to 16 b level or beyond at a S/H rate of several Mhz. So far however, the inability to control component matching to the same level, has limited their application to over-sampled converters [7]. The high over-sampling ratios used (usually above 100) limit the overall performance to audio-frequency conversion rates, where a digitally corrected pipeline could reach *the same accuracy at MHz rates.*

Nevertheless, the algorithm itself appears extremely robust against amplifier noise and nonlinearity. These effects have been included into high-level simulations, and the following conclusions seem to hold:

1) Noise of up to several *lsb* was found to have little impact on the final linearity, as long as every new weight value computed during the calibration process is obtained by averaging enough measurements.
2) A limited amount of amplifier nonlinearity (expressed as a "curvature" of the gain characteristic) does not influence the calculation of the weights much. This is because during the calibration, the amplifiers at the front of the pipeline are only biased at the low or high end of their range. Although the linearity criterion may reflect correct weights in the presence of such amplifier nonlinearity, the transfer characteristic of the converter will exhibit a smooth, composite curvature and associated INL errors. To keep the overall INL error below 1 *lsb*, the output-referred curvature of the gain in each amplifier must be kept below 1 *lsb*. By contrast, the effect of amplifier nonlinearity on the DNL is very limited.

XI. Conclusion

This paper provides a mathematical description of pipelined analog/digital converters, including the much-applied concept of "redundancy." Based on the analysis, a fully digital error correcting scheme is derived, based on a limited number of look-up tables. A powerful iterative algorithm ("accuracy bootstrapping"), in which the pipeline is used to measure its own errors, provides the necessary system identification to determine the table values. The amazing effectiveness of the method is demonstrated by a number of Monte-Carlo simulations of different pipeline architectures. Based on the known performance of analog blocks used in high-accuracy over-sampling converters, the new method is expected to reach similar accuracy at effective sampling rates that are two orders of magnitude higher.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of the Electrical Engineering Department at Texas A&M University, College Station, where this work was started around 1990. At Texas Instruments, they thank Dr. R. K. Hester, then manager of the Advanced Analog and CMOS Structures Branch and M. Chiang, manager of the Mixed Signal Design Center, for their personal help, and J. Hellums, M. De Wit, and S. Kiriaki for their technical assistance.

REFERENCES

[1] St. H. Lewis, "Video-rate analog-to-digital conversion using pipelined architectures," Ph.D. dissertation, Electron. Res. Lab., Univ. California, Berkeley, Memo. No. UCB/ERL M87/90, Nov. 18, 1987.

[2] B. Ginetti and P. Jespers, "A 1.5 Ms/s 8-bit pipelined RSD A/D converter," in *ESSCIRC Dig. Tech. Papers*, pp. 137–140, Sept. 1990.

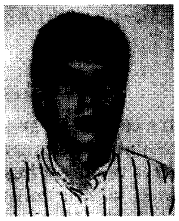
[3] E. G. Soenen, "Error modeling, self-calibration and design of pipelined analog to digital converters," Ph.D. dissertation, Dept. Elect. Eng., Texas A&M Univ., College Station, May 1992.

[4] E. G. Soenen and R. L. Geiger, "A fully digital self-calibration method for high resolution, pipelined A/D converters," in *Proc. 36th Midwest Symp. on Circuits Syst.*, Detroit, MI, Aug. 1993.

[5] J. Doernberg, H.-S. Lee, and D. A. Hodges, "Full-speed testing of A/D converters," *IEEE J. Solid-State Circuits*, vol. SC-19, no. 6, pp. 820–827, Dec. 1984.

[6] A. N. Karanicolas, H.-S. Lee, and K. L. Bacrania, "A 15 b 1Ms/s digitally self-calibrated pipeline ADC," *1993 IEEE Int. Solid-State Circuit Conf., Dig. Tech. Papers*, pp. 60–61, Feb. 1993.

[7] M. Rebeschini, N. R. Van Bavel, P. Rakers, R. Greene, J. Caldwell, and J. R. Haug, "A 16-b 160-kHz CMOS A/D converter using sigma-delta modulation," *IEEE J. Solid-State Circuits*, vol. 25, no. 2, pp. 431–436, Apr. 1990.



Eric G. Soenen was born in Brussels, Belgium on October 14, 1964. He received the degree of "Civil Electro-Mechanical Engineer" from the "Katholieke Universiteit Leuven," Belgium, in 1987. He received the M.S. degree in 1989 and the Ph.D. degree in 1992 from Texas A&M University, College Station, both in electrical engineering. His research included evaluation, error modeling, and self-calibration of ADC's, with a strong focus on pipelined architectures.

He joined Texas Instruments Inc., Dallas, in 1991. He began in the Power IC Group, and later transferred into the Mixed Signal Design Department. From August 1992 to March 1993 he took a leave of absence and moved back to Belgium to perform his military service. He is now back as a design engineer with Texas Instruments. Designs he was responsible for or has contributed to at the Mixed Signal Design Department include pipelined and successive approximation ADC's, remote-control IC's, speech synthesizers, and a custom controller core. He is currently working on a project in the video area.

Randall L. Geiger received the B.S. degree in electrical engineering and the M.S. degree in mathematics from the University of Nebraska in 1972 and 1973, respectively, and the Ph.D. degree in electrical engineering from Colorado State University in 1977.

He was a faculty member at Texas A&M University from 1977 to 1990. He joined the Electrical Engineering and Computer Engineering Department at Iowa State University in 1991 and currently serves as Professor and Chairman in the Department.

Dr. Geiger is a past Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, a past Circuits and Systems Society Editor of the Circuits and Devices Magazine, and a Past President of the Circuits and Systems Society. He is currently serving on the IEEE Periodicals Council and the IEEE Publications Board.