# An On-Chip DNL Estimation Technique and Reconfiguration for Improved Linearity in Current Steering Digital to Analog Converters

K.P.Sunil Rafeeque and V.Vasudevan

Department of Electrical Engineering

Indian Institute of Technology-Madras

Chennai-600036, India

Email: vinita@iitm.ac.in

The first version of this paper was sent to TCAS-II for consideration. The recommendation of the editor, the reviews and our replies to the comments of the reviewers are attached at the end of the paper. We have made extensive changes in the paper based on the reviews.

**Abstract**

This paper proposes a reconfigurable current steering digital to analog converter (DAC). The differential non-linearity error (DNL) of the DAC is estimated on-chip. This is used to reconfigure the switching sequence to get a lower integral nonlinearity error (INL). A 10 bit segmented DAC along with the associated circuits for DNL estimation was designed and fabricated using 0.35$\mu$m CMOS technology, through Europractice. The paper includes theoretical analysis, simulation and experimental results for the proposed technique.

## I. INTRODUCTION

Current steering digital to analog converters(DACs) are widely used in high frequency applications. The performance of the DAC depends on its linearity, which in turn is affected by random and gradient mismatches between the sources. The nonlinearities due to these mismatches are classified as differential nonlinearity and integral nonlinearity (INL) errors. The errors due to random mismatch can be reduced by an increase in area of the sources [1]. Careful layout techniques are used to minimize the effect of gradient errors [2].

In unary and segmented DACs, the INL is of particular concern [3]. The INL in the DAC depends on the number of consecutive current sources that have the same error polarity. This occurs usually due to gradient errors. However, in a particular manufacturing run, random errors could also result in a large INL in some of the dies. The INL depends, quite strongly, on the order in which the current sources are switched. With some knowledge of the gradient error, this can be optimized. Algorithms such as $Q^2$ random walk [4, 5], a gradient-error and edge effect tolerant switching scheme [6], Sort and Group Algorithm and INL Bounded Algorithm [7], have been proposed to arrive at the optimum switching sequence. The $Q^2$ random walk uses an error profile from a test chip to determine the switching sequence. The other two algorithms have been arrived at after simulations in the presence of linear and quadratic gradient errors. In all these cases, a switching sequence that is obtained from simulations/test runs is used in the design and layout of the DAC. Once the DAC is fabricated, the order in which the sources are switched cannot be changed.

There are several advantages to optimizing the switching sequence after the DAC has been fabricated. As in the previous case, it can be used to compensate for gradient mismatches. It can further improve performance by compensating for errors due to random mismatch. The random

nature of the mismatch occasionally produces mismatches greater than $0.5LSB$. A post-silicon switching sequence optimization can eliminate the effect of such errors if redundant cells are included. A flexible switching sequence can also be used to pre-distort the transfer function. The "programmable" decoder allows for much simpler layout and routing, especially useful in high resolution DACs. However, changing the switching order is much more useful if the error estimation and generation of a switching sequence is done automatically.

In this paper, we explore the possibility of using an on-chip error estimation technique to optimize the switching sequence for a lower INL. This is studied analytically, through simulations and experimentally. Most high resolution DACs have a segmented architecture, with the INL contributed mostly by the upper segment sources. We propose simple circuits that can be used to estimate error bounds for each upper segment current source, sort the current sources and generate a switching sequence that results in a lower INL. This sequence is stored in registers and used to switch the appropriate current source at each code. It can be also used to replace faulty cells, if the DAC has redundant cells. Unlike [8], we do not perform any calibration. Since we are not doing any calibration, it is not possible to alter the DNL of the DAC. Our aim is to optimize the switching sequence based on on-chip DNL measurements, so that a better INL can be obtained. This has not been attempted previously. Alternatively, we show that it is possible to reduce the area of the current sources for a given yield.

This paper is organized as follows. In section II, conventional switching schemes are discussed. Section III includes the proposed post silicon switching sequence optimization scheme along with analytical and simulation results. The on-chip DNL estimation is discussed in VII. The circuits used to estimate error bounds for each current source, sorting and reconfiguration logic are discussed in section VIII. Section XI contains experimental results from silicon samples fabricated using $0.35\mu$ CMOS technology, through Europractice. Section XII contains the conclusions.

## II. Conventional switching schemes

In a current steering DAC, the current sources are arranged in square matrix and the effect of gradient mismatch on the INL is reduced by ordering the switching sequence. Different switching sequence schemes have been proposed. These schemes are briefly reviewed and compared with the classical scheme in which current sources are switched row by row.

### A. Row-Column Switching Scheme

In Row-Column switching scheme [3, 9–11], the gradient errors are averaged out in two directions independently. In this sequence, linear gradient errors are canceled by a diametrically opposite source with error of opposite polarity. This scheme reduces the error due to linear gradient but is not effective for quadratic errors [7].

### B. Hierarchal symmetric Switching Scheme ($Q^2$ Random walk)

In this scheme [4, 5] whole matrix is divided in to coarse and fine regions. The coarse matrix switching is used to randomize quadratic gradient errors and fine matrix switching compensates for local linear gradient errors. The scheme uses an error profile from a fabricated sample to optimize the sequence and is tested on an 8-6 segmented DAC. It provides very good compensation for quadratic gradient and reasonable compensation for linear gradient errors, at the cost of complex interconnect.

### C. Gradient-Error and Edge Effect Tolerant Switching Scheme

In this scheme [6] coarse current sources are built multiple element taken from different locations of a two dimensional matrix. The element selection enable to a reduction in INL growth. A good reduction in INL is obtained at the cost of complex interconnect.

### D. Sort and Group scheme [7]

In Sort and Group, current sources are sorted and arranged in ascending or descending order. The sorted sources are then re-grouped in several ways to reduce the INL. They are good for linear gradient errors, but it is not clear how well they perform in the presence of quadratic errors.

### E. INL Bound Switching Scheme [7]

The sources are switched in such a way that the maximum INL is always within the specified range. The INL bound is chosen to be between the lower bound (i.e half the maximum DNL) and a value obtained from the sort and group scheme. It gives a low value for the INL, but implementing it in a post-silicon optimization scheme will require accurate measurements to be done on-chip and hence a large area overhead.

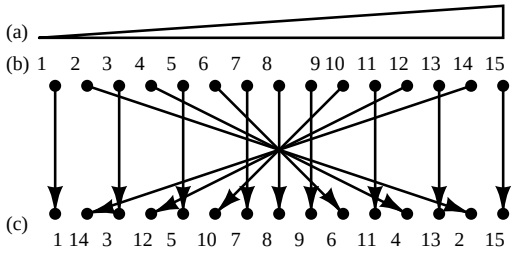


Fig. 1. Switching sequence mapping algorithm. (a) Error after sorting in ascending order (b) sequence of current source after sorting in ascending order and (c) sequence of current source after mapping.

## III. Post-Silicon Switching Sequence Optimization Scheme

Switching sequence optimizations are used only for high resolution DACs (10-14 bits), which are usually segmented DACs [2–4,9–11]. They are generally used to compensate for the gradient mismatch in the upper segment sources. It is assumed that the error distribution is known before fabrication. They cannot be used for compensating for errors due to random mismatch. The designer has to ensure that the random mismatch is less than the limiting case by increasing the area.

Our scheme is a post-silicon switching scheme that can take care of both types of errors. It is a variation of the sort and group scheme and can be implemented with a relatively low area overhead.

### *A. Switching Sequence Optimization Algorithm*

The first step is to estimate the DNL at each code. The strategy employed here is to compensate every positive DNL with a negative DNL. Based on the results of the on-chip DNL estimation, the current sources are first sorted in the ascending order of DNLs. The sources are then mapped as shown in Figure 1. The current sources that have the maximum and minimum DNL remain at the two ends. The switching order for the two sources next to the maximum and minimum is interchanged. This process is continued for all sources. Effectively, all the “odd numbered” sources in the sorted array remain in the same position and all the “even numbered” sources in the top half are swapped the corresponding sources in the lower half of the sorted array.

## IV. ANALYSIS

### A. Random Errors

*1) Unsorted Array:* Since the focus is on improving the INL, we look at the absolute value of the INL in an $N$ bit DAC. Assume that current due to each cell j, $I_j$, can be written as

$$I_j = I_{unit} + X_j \tag{1}$$

$I_{unit}$ is the current corresponding to $1\ LSB$ and $X_j$ is the error in the current. It is assumed that the errors $X_j$ are normally distributed with zero mean and standard deviation $\sigma$, i.e $N(0, \sigma)$. Further it is assumed that the error in each current source is uncorrelated with the errors in other sources. This limits the errors considered to random errors. All systematic errors like gradient errors and biasing errors are excluded. The gain error is, by definition, the error in the output current corresponding to the digital code $2^N - 1$. This is given by

$$G = I_{2^N-1} - (2^N - 1)I_{unit} = \sum_{i=1}^{2^N-1} X_i \tag{2}$$

The gain error $G$ is once again a random variable. Since it is a sum of $2^N - 1$ zero mean normally distributed random variables, G is also a normally distributed random variable with zero mean and standard deviation equal to $\sigma\sqrt{2^N - 1}$ i.e. $N(0, \sigma\sqrt{2^N - 1})$. The INL at each code has be estimated after correcting for the gain error. The INL at code $j$ is therefore obtained by subtracting out $j$ times the average error ($G/2^{N-1}$) from the total error at code $j$. Therefore, after correcting for the gain error, the error in the current at each code $j$ or the equivalently the INL at code $j$, $INL_j$, is a random variable given by:

$$INL_j = \sum_{i=1}^{j} X_i - \frac{j}{2^N - 1} \sum_{i=1}^{2^N-1} X_i \tag{3}$$

As expected the INL is zero at the two end points. Note that, the same result can be obtained using the following procedure. First find the average current ($I_{av}$). The error in the current at code $j$ is thus $I_j - I_{av}$. This is the same as the DNL at code $j$. The INL at code $j$ can then be obtained by summing up the DNLs upto code $j$.

The variance of $INL_j$, $\sigma^2_{INL_j}$ can be calculated as:

$$\begin{aligned} \sigma^2_{INL_j} &= E\{INL_j^2\} \\ &= j\sigma^2 - \frac{j^2\sigma^2}{2^N - 1} \end{aligned} \tag{4}$$

The maximum value of $\sigma^2_{INL_j}$ can be found by setting the first derivative of equation (4) with respect to $j$ to zero. The maximum standard deviation is given by $\sigma_{INL_{max}} = 0.5\sigma\sqrt{2^N - 1}$. Substituting this maximum value in equation (4), we can find the value of $j$ at which this occurs. It can be easily verified that the maximum value occurs at midcode i.e at $m = 2^{N-1}$ or at $m = 2^{N-1} - 1$. Figure 2 shows a plot of the standard deviation of the INL at various codes for a 10 bit and 14 bit DAC. The percentage mismatch in the two cases was assumed to be the values given in the table I.

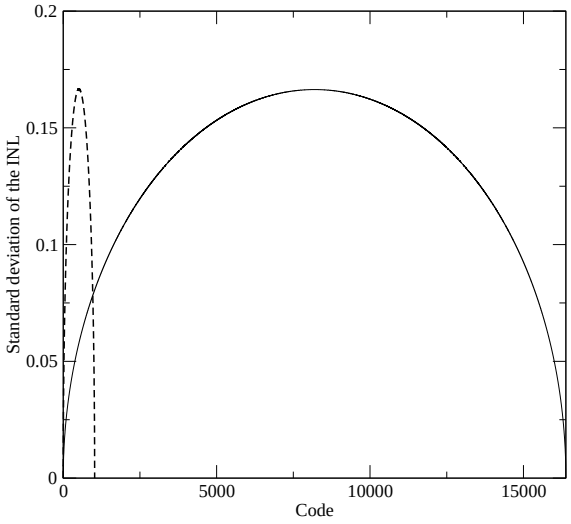


Fig. 2. The standard deviation of the INL at various codes for a 10 and 14 bit DAC

The INL at each code is a normally distributed random variable. Therefore, the maximum INL can occur at any code. However, a reasonable assumption is that it is more likely to be determined by the distribution that has the largest standard deviation. If we assume that the $3\sigma_{INLmax}$ value determines the INL yield ($|INL|_{max} <$ 0.5LSB), the percentage mismatch that can be tolerated for DACs with various resolutions is given in Table I. The $3\sigma_{INL_{max}}$ will essentially ensure approximately 99.7% yield. The results for a 14 bit DAC are quite close to the numbers computed in [12]. However, it is a more aggressive estimate as compared to the values in [2, 13].

After gain correction, the DNL at code $j$ is given as:

$$DNL_j = X_j - \frac{1}{2^N - 1} \sum_{i=1}^{2^N - 1} X_i \tag{5}$$

| N | mismatch ($\frac{\sigma_I}{I}\%$) |
|---|---|
| 8 | 2.08 |
| 9 | 1.47 |
| 10 | 1.04 |
| 12 | 0.52 |
| 14 | 0.26 |

TABLE I

THE PERCENTAGE MISMATCH ERROR FOR 99.7% YIELD FOR VARIOS RESOLUTIONS

The variance of $DNL_j$ is thus given by:

$$\begin{aligned} \sigma^2_{DNL_j} &= \sigma^2 - \frac{\sigma^2}{2^N - 1} \\ &\approx \sigma^2 \end{aligned} \tag{6}$$

Therefore, the effect of the gain error on the DNL is negligible.

In a segmented DAC, the INL is mainly due to the upper segment sources. This is almost the same as that for the unary DAC and the above equations can be used. This happens because the upper segment sources are usually implemented by connecting the appropriate number of unit size current sources in parallel. However, at the transitions from the lower to the upper segment, the DNL is expected to be larger. If there are $2^M - 1$ current sources in the lower segment, it effectively means that each upper segment source consists of $2^M$ sources connected in parallel. Therefore, the standard deviation of the error in the upper segment source is $\sigma\sqrt{2^M}$. Since all the lower segment sources are turned off every time an upper segment source is turned on, the standard deviation of the error (and $DNL_{seg}$) at the transitions can be calculated as [13].

$$\sigma_{DNL_{seg}} = \sigma\sqrt{2^{M+1} - 1} \tag{7}$$

*2) Sorted Array:* If the current sources are sorted according to the algorithm described, the mapping produces the negative and positive maximum DNL at the two ends and a minimum in the middle. Therefore, minimum INL that can be obtained using this scheme is the same as the maximum DNL, either the most negative or the most positive value. The expected value of the minimum/maximum DNL can be obtained using order statistics. If $n$ samples from a standard

normal distribution are arranged in the ascending order of magnitude, $x_1$, $x_2$ ..., $x_n$, then the $i^{th}$ value of the set, $x_i$, is called the $i^{th}$ normal order statistic. The expected value of $x_1$, which is the minimum value, can be written as [14]:

$$E\{x_1\} = n \int_{-\infty}^{\infty} x f(x)[1 - F(x)]^{n-1} dx \tag{8}$$

where $f(x)$ is $N(0,1)$ and $F(x)$ is the corresponding cumulative distribution function. The expected value of $x_n$, which is the maximum, is $-E\{x_1\}$. The expected value of the $i^{th}$ order statistic for various values of $n$ have been tabulated in [14]. It is seen from these tables that the $E\{DNL_{max}\}(= -E\{DNL_{min}\}$ is about $2\sigma$ when $n = 32$. It increases towards $3\sigma$ for higher sample sizes.

After sorting in the ascending order and rearranging, the lower bound for $E\{|INL|_{max}\}$ is given by the $E\{|DNL|_{max}\}$. Simulations show that it is of the $3\sigma$ for $n \geq 16$. This is considerably lower than for the unsorted array. Therefore, with sorting, it is possible to improve the INL yield for a given area of the current sources. Alternatively, for a given yield, it is possible to have a much lower area for the current sources. This is discussed in more detail in section VI. Note that, since we are not doing any calibration, no change in the DNL is possible. Only the INL is reduced by reordering the switching sequence.

In a bit segmented DAC, the DNL is dominated by the errors at the transition points. The DNL at these points is given by

$$DNL_i = X_{u_i} - Y \tag{9}$$

Here $X_{u_i}$ is the error in the $i^{th}$ upper segment current source and $Y$ is the sum of the errors in the lower segment sources. Based on the DNL at these points, the upper segment sources can be sorted in the ascending order. Therefore, the expected value of the minimum DNL is $E\{X_{u_1} - Y\} = E\{X_{u_1}\}$ (since the lower segment errors are distributed according to $N(0,\sigma)$). Thus after sorting, the $E\{|DNL|_{max}\}$ will depend only on the errors in the upper segment sources. Following the previous discussion on sorted unary arrays, the $E\{|INL|_{max}\}$ is about $3\sigma_u = 3\sigma\sqrt{2^M}$ where $M$ is the number of lower segment bits.

### *B. Gradient Errors*

The two types of gradient errors commonly encountered are linear and quadratic gradient errors. First, we assume that the current sources have a linear gradient error and the value of the

current can be written as $I + \Delta$, $I + 2\Delta$, ..., $I + (2_N - 1)\Delta$. Following a procedure similar to that for random errors, the DNL at code 'j' after gain error correction can be written as:

$$DNL_j = j\Delta - (2^{N-1} - 1)\Delta \tag{10}$$

Therefore, the DNL is initially negative, it becomes zero around the midcode value and then becomes positive. Since the distribution of the positive and negative errors is almost the same, after sorting each positive DNL is compensated by a negative DNL of almost the same magnitude. Hence the $E\{|INL|_{max}\}$ will be very close to $E\{|DNL|_{max}\}$. This is seen in table III.

We now assume a quadratic gradient error. Here we assume the following error distribution. The error in the current sources $2^{N-1}$ (midcode) is '0' and the error increases quadratically in both directions. The values of the currents can be written as $I + (2^{N-1})\Delta$, ..., $I + 4\Delta$, $I + \Delta$, I, $I + \Delta$, $I + 4\Delta$, ..., $I + (2^{N-1})\Delta$. Once again after subtracting out the gain error, the DNL at code 'j' can be written as

$$DNL_j = DNL_{2^N - j} = (2^{N-1} - j)^2\Delta - \frac{2^{N-1}(2^{N-1} - 1)\Delta}{3} \tag{11}$$

It can be easily verified that the sum of all DNL values is zero. The DNL is zero at codes

$$j = 2^{N-1} - \sqrt{\frac{2^{N-1}(2^{N-1} - 1)}{3}} \tag{12}$$

and

$$j = \sqrt{\frac{2^{N-1}(2^{N-1} - 1)}{3}} \tag{13}$$

For a 10 bit unary DAC, this occurs approximately at codes 217 and 809. The most positive value of the DNL occurs at the two ends and is given by

$$DNL_{max} = (2^{N-1} - 1)^2 - \frac{2^{N-1}(2^{N-1} - 1)}{3} \tag{14}$$

The most negative value occurs at midcode and is given by

$$DNL_{min} = -\frac{2^{N-1}(2^{N-1} - 1)}{3} \tag{15}$$

Clearly, the most positive value is almost double that the most negative value. Therefore, if sorted in the ascending or descending order the errors cannot compensate for each other well.

Thus in the presence of quadratic error, the $E\{|INL|_{max}\}$ will be larger than the lower bound of $E\{|DNL|_{max}\}$. This is also seen in simulations shown in table III.

The method obviously works very well when the distribution of positive and negative errors is more or less even. This condition is more or less satisfied when linear gradient and random errors are dominant. In the case of zero mean quadratic gradient error, the magnitudes of the positive and negative errors are not very well balanced. Therefore, there are uncompensated errors after each pair, contributing to the INL.

### *C. Effect of finite Measurement resolution*

In practical systems, the efficiency of this sorting scheme also depends on the resolution of the measurement used to estimate the DNL. Due to area constraints, this will be limited. As a result there are inaccuracies in sorting, resulting in uncompensated errors. Assume that the DNL is measured using an M-bit ADC with full-scale voltage of $\pm 1LSB$. This implies a step size of $\Delta = 2LSB/2^M$ and a quantization error variance of $\Delta^2/12$. Therefore, there is an error in the DNL estimation and the standard deviation of this error is $\sigma_q = \frac{\Delta}{\sqrt{12}}$. Assume that at each code $j$, there is an error $q_j$ in the DNL due to the quantization error. The error in the INL at each code $j$ due to the quantization error can be written as:

$$INL_{qj} = \sum_{i=1}^{j} q_i \tag{16}$$

If we do an analysis similar to that for the INL of the unsorted array, the maximum standard deviation of the INL due to the quantization error, $\sigma_{INL_q}$ is given as:

$$\sigma_{INL_q} = 0.5\frac{\Delta}{\sqrt{12}}\sqrt{2^N - 1} \tag{17}$$

A reasonable assumption is that error in the expected value of the maximum INL after sorting is of the order of $\sigma_{INL_q}$. Therefore:

$$E\{|INL|_{max}\} = E\{|DNL|_{max}\} + 0.5\frac{\Delta}{\sqrt{12}}\sqrt{2^N} \tag{18}$$

## V. Simulation results

A large number of simulations with different error variances and quantizer resolution were done to study the effect of this scheme. Since the scheme is meant for segmented DACs, the

simulations were carried out for two cases - a 5+5 and 8+6 bit segmented DAC. The simulations were done as follows. In each simulation of the 5+5 bit case, 31 random numbers were generated using $N(0, \sigma)$ for the lower segment errors. For the upper segment, errors were generated using $N(0, \sigma\sqrt{31})$. The DNL and INL at each code was computed after compensating for the gain error. Using this $|DNL|_{max}$ and $|INL|_{max}$ is obtained for each simulation. The average of 250 simulations is reported in table II. A similar procedure is followed for gradient errors, except that the errors are not generated randomly.

| 5+5 bits segmented DAC | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Before optimization | | | | After optimization | | | | | | | | |
| | $\|DNL\|_{max}$ | | $\|INL\|_{max}$ | | $\|INL\|_{max}$ for different measurement resolutions | | | | | | | | |
| | | | | | $\infty$ -bit | | | 5-bit | | | 3-bit | | |
| | (Sim) | | (Sim) | | (Sim) | | (Anal) | (Sim) | | (Anal) | (Sim) | | (Anal) |
| $\sigma_{I_L}$ (LSB) | Mean (LSB) | Yield (%) | Mean (LSB) | Yield (%) | Mean (LSB) | Yield (%) | Mean (LSB) | Mean (LSB) | Yield (%) | Mean (LSB) | Mean (LSB) | Yield (%) | Mean (LSB) |
| 0.0295 | 0.491 | 61 | 0.816 | 3 | 0.488 | 60 | 0.468 | 0.4686 | 60 | 0.5182 | 0.61 | 18 | 0.668 |
| 0.0208 | 0.3424 | 93.2 | 0.5878 | 31 | 0.3474 | 96.4 | 0.33 | 0.3373 | 98 | 0.380 | 0.4585 | 67 | 0.531 |
| 0.0147 | 0.2473 | 100 | 0.42 | 74.4 | 0.2458 | 100 | 0.234 | 0.2527 | 99.6 | 0.284 | 0.3227 | 100 | 0.434 |
| 0.0104 | 0.1737 | 100 | 0.2893 | 98.4 | 0.1717 | 100 | 0.165 | 0.1919 | 100 | 0.215 | 0.2295 | 100 | 0.365 |
| 8+6 bits segmented DAC | | | | | | | | | | | | | |
| 0.0074 | 0.2156 | 100 | 0.8422 | 6 | 0.2444 | 100 | 0.2502 | 0.3910 | 82.4 | 0.3942 | 0.6880 | 33 | 0.8264 |
| 0.0052 | 0.1521 | 100 | 0.5942 | 35 | 0.1724 | 100 | 0.1758 | 0.3357 | 95.6 | 0.3200 | 0.4856 | 58 | 0.7520 |
| 0.0037 | 0.1076 | 100 | 0.4205 | 75.6 | 0.122 | 100 | 0.1251 | 0.2804 | 96.4 | 0.2691 | 0.3437 | 80.4 | 0.6982 |
| 0.0026 | 0.0761 | 100 | 0.2971 | 98.8 | 0.0862 | 100 | 0.0879 | 0.2168 | 100 | 0.2319 | 0.2428 | 97.2 | 0.6641 |

TABLE II

IMPROVEMENT IN INL AFTER SWITCHING SEQUENCE OPTIMIZATION WITH RANDOM MISMATCH IN CURRENT SOURCES. THE STANDARD DEVIATION OF LSB SOURCES, CORRESPONDING MEAN OF $|DNL|_{max}$ AND $|INL|_{max}$ AND YIELD FROM 250 SIMULATIONS

It is seen that the results of the simulation match reasonably well with analytically predicted values. Since it is statistical, the match is better for larger number of current sources and lower $\Delta$ values. It is also apparent from the table that a measurement resolution of 5 bits is sufficient to get close to the lower bound and a resolution of 3-bits is sufficient to get the mean value of the maximum INL to less than $0.5LSB$. Figure 3 shows the histogram of the maximum DNL and INL in the 10 bit case with $\frac{\sigma I_L}{I_L} = 0.0208$.

| 8+6 bits segmented DAC | | | | | |
|---|---|---|---|---|---|
| | Before optimization | | After optimization | | |
| Grad: Errors | $\|DNL\|_{max}$ | $\|INL\|_{max}$ | $\|INL\|_{max}$ for different measurement resolutions | | |
| | | | $\infty$ -bit | 5-bit | 3-bit |
| Lin: Grad: | 0.3 | 10 | 0.3 | 0.4 | 0.5 |
| Quad: Grad: | 0.24 | 3.8 | 0.55 | 0.8 | 3.2 |

TABLE III

IMPROVEMENT IN INL AFTER SWITCHING SEQUENCE OPTIMIZATION WITH GRADIENT MISMATCH IN CURRENT SOURCES. THE STANDARD DEVIATION OF LSB SOURCES, AND CORRESPONDING MEAN OF DNL AND INL FROM 250 SIMULATIONS

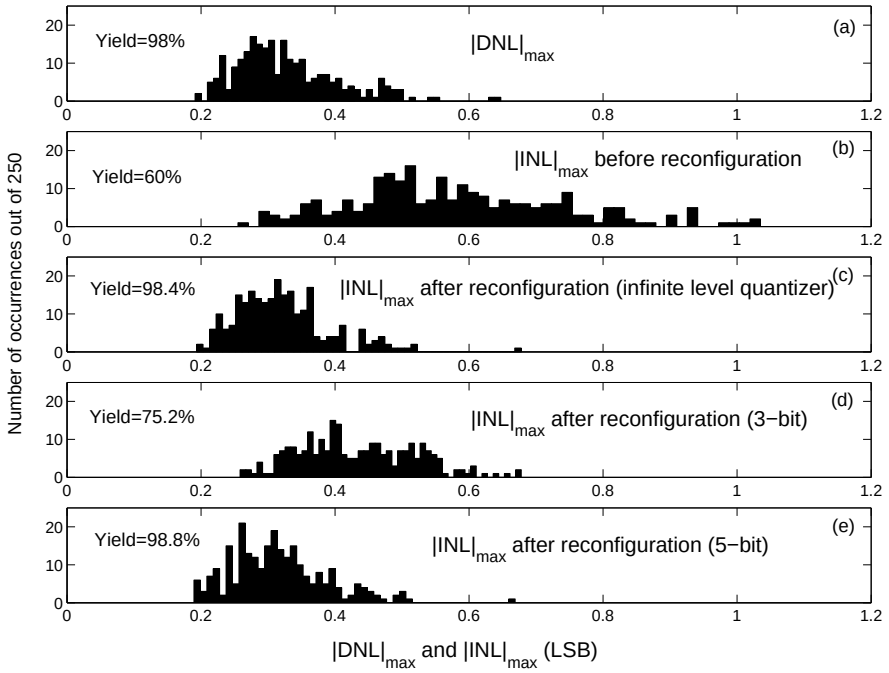


Fig. 3. Histogram of $|DNL|_{max}$ and $|INL|_{max}$ of a 5+5 bit DAC with $\frac{\sigma I_L}{I_L} = 0.0208$ for 250 simulations. (a) $|DNL|_{max}$. (b) $|INL|_{max}$ before reconfiguration. (c) $|INL|_{max}$ after switching sequence optimization(infinite level quantization). (d) $|INL|_{max}$ after switching sequence optimization(with 3-bit quantization levels). (e) $|INL|_{max}$ after switching sequence optimization(with 5-bit quantization levels).

The yield ($|\text{INL}|_{max} < 0.5LSB$ )was about 60% with original sequence. With a measurement resolution of 5-bits, reconfiguration enhances the yield($|\text{INL}|_{max} < 0.5LSB$) to 98.8% . With 3-bit resolution the yield increases to 75.2%. The table III shows that the scheme is very effective with linear gradient errors and reasonably effective with quadratic gradient errors, as expected.

## VI. Effect of optimization on the area of the current sources

According to the Pelgrom model [1], we have

$$\frac{\sigma_I^2}{I^2} \propto \frac{1}{WL} \tag{19}$$

Supposing we use a five bit measurement resolution to sort the upper segment sources of a 5+5 bit segmented DAC. Then $E\{|INL|_{max}\}$ after optimizing the switching sequence is become $E\{|DNL|_{max}\} + 0.051$LSB, i.e. $|INL|_{max}$ increased by $0.051$LSB from it's infinite quantization value. Therefore the $\frac{\sigma_{I_l}}{I_L}$ for $E\{|DNL|_{max}\} = 0.449$ is about 0.0282. This does not guarantee 100% yield. Simulations shows that, this $\frac{\sigma_{I_l}}{I_L}$ produces an yield of 3% without reconfiguraton and 60% with reconfiguration. This corresponds to a unit cell area of 7-bit DAC. Further simulations with $\frac{\sigma_{I_l}}{I_L}$ of 0.0208, 0.0147 and 0.0104 shows an yield for 31%, 74.4% and 98.4% without reconfiguration 98%, 99.6% and 100% with reconfiguration respectively. This indicate that with the help of reconfiguration, current source area can be reduces to half of the conventional scheme without compromising on yield.

Similarly, in 8+6 bit DAC with 5-bit quantization, the growth in INL due to quantization error is about $0.203$LSB. In this case also simulation shows that unit cell area can be reduces to half of the conventional scheme without compromising on yield. This is also seen in simulations shown in table II. A 10 bit DAC with 8-bit unit cell area gives about 93% yield(limited by DNL). Simmilarly a 14-bit DAC with 12-bit unit cell area gives about 95.6% yield(limited by INL).

## VII. Outline of the technique used to estimate DNL

The proposed scheme was evaluated using a 10-bit segmented DAC with 5-bits in each segment. As seen from the simulations, a 3 bit measurement resolution is very effective in improving the INL. This is also seen from figure 4, where the INL yield improves from $85\%$ to $97\%$. We have therefore designed measurement circuits that have 3 bit(0.25LSB) resolution. The results of the measurement are used to optimize the switching sequence of the upper segment sources.

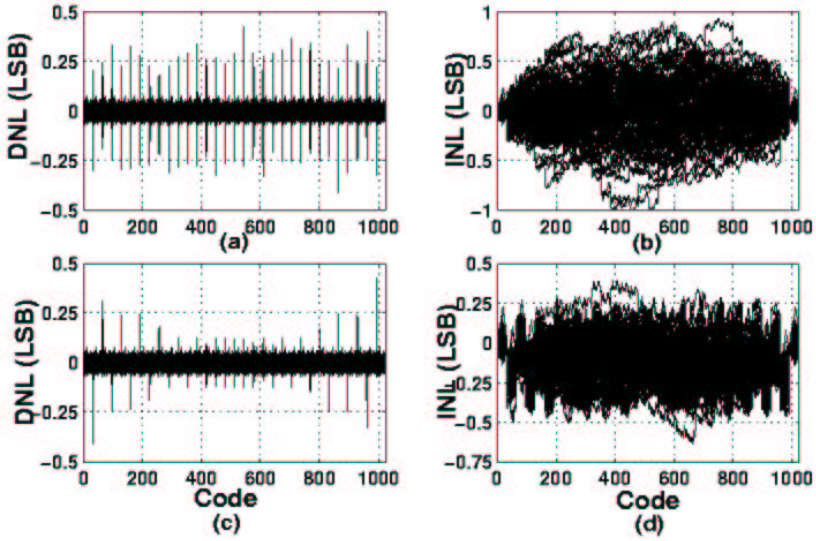


Fig. 4. DNL and INL from 100 MATLAB simulations for a 5+5 bit segmented DAC. (a) and (b) corresponding to the raw random sequence. (c) and (d) corresponding to modified sequence sorted using estimated DNL with $0.25LSB$ resolution. Note that the figure contains results from 100 simulations, so it appears that there are more than 32 transition points.

Accurate measurement of output voltage at different code requires an ADC of linearity better than that of the DAC. This requires both large area and power. As an alternative to this, we measure the step size corresponding to two adjacent codes. Here the output corresponding to the first code is sampled on capacitors and subtracted from the voltage corresponding to the next code. This provides the step size. All subsequent measuring circuits process only the step size and hence require only a very small dynamic range as compared to the entire DAC output. The only requirement of the scheme is to have linear on-chip capacitors, which are generally available.

An offset compensated voltage comparator along with the two reference current steering cells of 1 LSB and 0.5 LSB forms the 'sample and subtract' and digitization of the DNL up to 0.25 LSB resolution. The two reference current steering cells are merged with DAC such a way that it can be added with DAC output in any fashion to have the required sequence for proposed DNL estimation scheme. The block schematic of the measurement scheme is shown in figure 5. The comparator samples the differential output voltage of the DAC at $t_1$ and compares it with output at time $t_2$. If $V(t_1)$ and $V(t_2)$ are the two voltages corresponding to code $M$ and

$N$, the output of the comparator ($C$) is given as:

$$\begin{aligned} C &= 1, \ V(t_2) - V(t_1) > 0 \\ &= 0, \ V(t_2) - V(t_1) < 0 \end{aligned}$$

Let $C_a$ be the comparator output with the two inputs interchanged. Ideally $C$ will be complement of $C_a$. In practice, the same output could be obtained if the difference is within the residual offset of the comparator. The voltage ($V_N$) corresponding to each code $N$, is compared with $V_{N-1}$, $V_{N-1} + .5LSB$, $V_{N-1} + 1\ LSB$ and $V_{N-1} + 1.5LSB$. An estimate of DNL is obtained as follows. If the outputs are same in both the tests, the two voltages are considered equal, to within the comparator resolution. Otherwise $V_N$ is computed as mean of the upper and lower bound. A lookup table is used to estimate the DNL from these four pairs of tests. Table IV shows the possible comparator outputs and corresponding DNL error. When the switch "S" is ON the DAC output is stored in the two capacitors. When "S" is OFF, the difference between DAC output and stored voltage is fed to the comparator. The switches $T_1$ and $T_2$ are used to interchange the inputs of the comparator.

## VIII. Circuit Design

A 10-bit current steering DAC with a programmable decoder and some simple circuits to estimate the DNL were designed in $0.35\mu m$ AMS CMOS process and fabricated through Europractice. The circuits to estimate the DNL include an offset compensated comparator, a state machine that controls the DNL estimation cycle and some digital logic to process the results of the comparator. The comparator is implemented on-chip. The controller and the logic required to determine the DNL and switching sequence using the results of the comparison were implemented on an external FPGA. The DAC was designed to operate in two modes

1) Normal mode:- In this mode, it works as normal 10 bit segmented DAC addressed by the output of the thermometer decoder.
2) Test and reconfigure mode:- In this mode, the thermometer decoder is bypassed and each cell can be accessed independently. The DAC can then be addressed according to the switching sequence determined from measured DNL. It also allows us to compare each current steering cell with any of the other cells.

| Comparator outputs( $C_aC$) | | | | |
|---|---|---|---|---|
| Test 1 | Test 2 | Test 3 | Test 4 | DNL |
| 01 | | | | DNL $<$ -1 LSB |
| 00 | | | | DNL = -1 LSB |
| 11 | | | | DNL = -1 LSB |
| 10 | 01 | | | DNL = -0.75 LSB |
| 10 | 00 | | | DNL = -0.5 LSB |
| 10 | 11 | | | DNL = -0.5 LSB |
| 10 | 10 | 01 | | DNL = -0.25 LSB |
| 10 | 10 | 00 | | DNL = 0 LSB |
| 10 | 10 | 11 | | DNL = 0 LSB |
| 10 | 10 | 10 | 01 | DNL = 0.25 LSB |
| 10 | 10 | 10 | 00 | DNL = 0.5 LSB |
| 10 | 10 | 10 | 11 | DNL = 0.5 LSB |
| 10 | 10 | 10 | 10 | DNL >= 0.5 LSB |

TABLE IV

POSSIBLE SEQUENCE OF COMPARATOR OUTPUTS($C_aC$) AND THE CORRESPONDING DNL ERROR. $V_N$ IS COMPARED WITH $V_{N-1}$, $V_{N-1}+0.5LSB$, $V_{N-1}+1\ LSB$ AND $V_{N-1}+1.5LSB$ IN THE FOUR TESTS TEST1, TEST2, TEST3 AND TEST4.

The following sections outline the design of the DAC, comparator and the logic for reconfiguration.

### *A. DAC*

A schematic of the analog circuit including the current steering DAC is shown in Figure 5. It consists of 31 current steering cells in upper and lower segments. The lower segment current sources are designed with an accuracy of $0.01LSB$. The accuracy of upper segment sources limited to $0.37LSB$. This will introduce large error in INL. The scaling upper segment will introduce a systematic offset in the upper segment currents. This is corrected using an current tuning loop [15]. A $0.01LSB$ accuracy of lower segment current sources is necessary for the accurate estimation of DNL using the proposed measurement scheme.

Two extra $one\ LSB$ cells are provided in the lower segment for DNL estimation. The second extra cell is converted into a $half\ LSB$ cell by connecting one arm directly to $V_{dd}$ as shown

in figure 5. This cell is therefore, not fully differential and gives half the voltage difference as compared to the other $one\ LSB$ cells. These two cells along with the comparator and digital logic effectively form a two bit analog to digital converter (ADC). However, since we measure the difference in the voltage corresponding to two adjacent codes, it is equivalent to measuring the output voltage using a 12 bit ADC and then finding the difference.

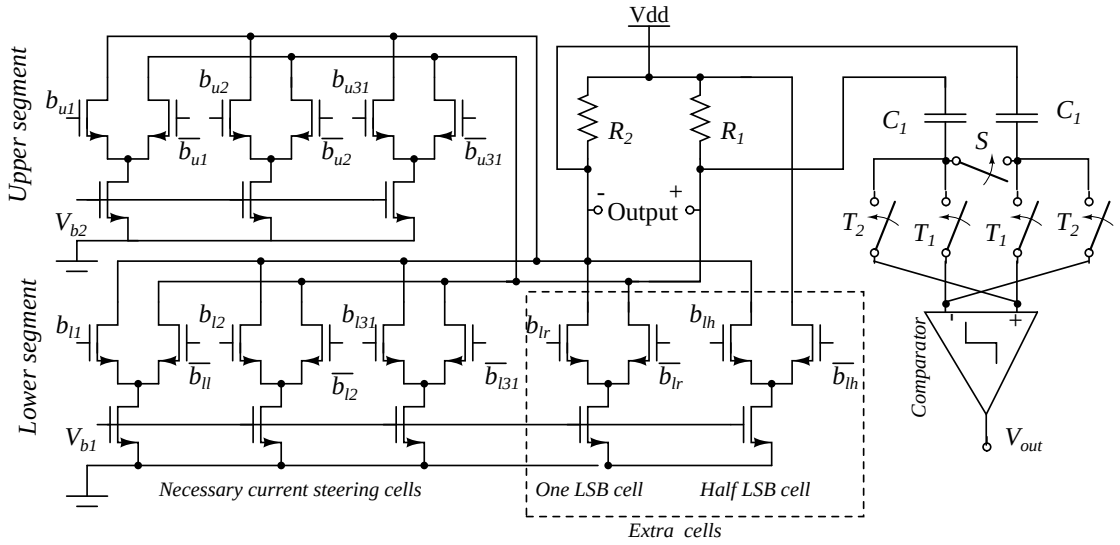


Fig. 5. Schematic of the segmented current steering architecture and error quantizer. Extra one LSB and half LSB cells have been added for error quantization

### *B. Design of the comparator*

The comparison consists of a fully differential offset cancellation network, a pre-amplifier, Schmitt trigger and level shifter. Figure 6 shows the schematic of the comparator. Here $\phi_1$ and $\phi_2$ refer to the two phases of clock signal. The output offset storage technique [16] is used to correct for the offset voltage of the preamplifier. This was preferred over the input offset compensation scheme, since the design of the pre-amplifier becomes much simpler. Capacitors $C_{o1}$ and $C_{o2}$ have been added for this purpose.

During $\phi_1$, capacitors $C_{i1}$ and $C_{i2}$ store the $V_1(\phi_1) - V_{ref}$ and $V_2(\phi_1) - V_{ref}$ respectively. During $\phi_2$, capacitors $C_{i1}$ and $C_{i2}$ come in series with $V_1$ and $V_2$. Therefore, the output of the amplifier is given by:

$$\Delta V = V_1(\phi_2) - [V_1(\phi_1) - V_{ref}] - (V_2(\phi_2) - [V_2(\phi_1) - V_{ref}]) = [V_1(\phi_2) - V_2(\phi_2)] - [V_1(\phi_1) - V_2(\phi_1)] \quad (20)$$

This small voltage $\Delta V$, is converted to a digital signal using a Schmitt trigger. The Schmitt trigger is enabled with delayed version of $\phi_2$ $(\phi_{2d})$ to allow for charging of its parasitic input capacitance. The maximum speed of the circuit depends on the two time constants involving the two switched capacitor networks. The amplifier topology is shown in figure 7. The Schmitt trig-

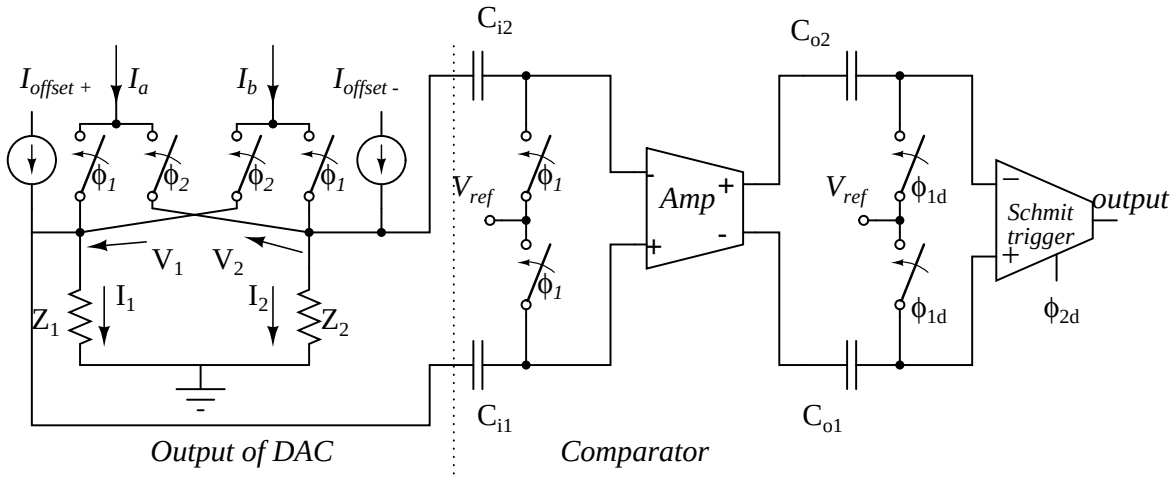


Fig. 6. Comparator used for DNL estimation

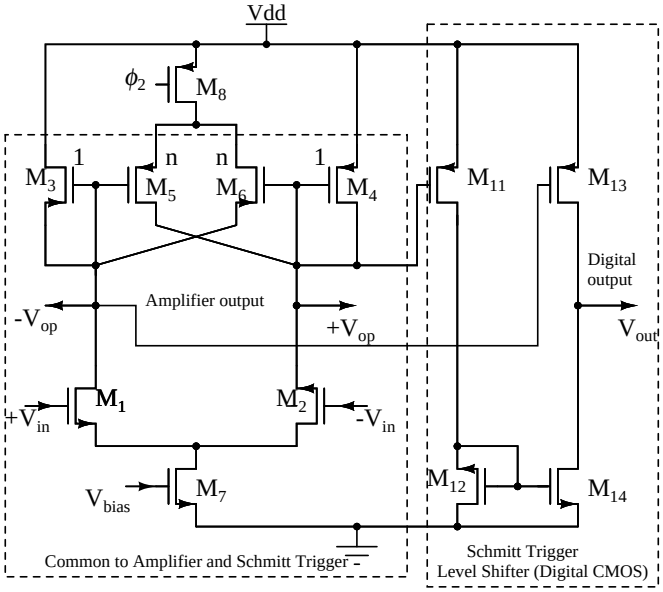


Fig. 7. Schematic of pre-amplifier(Amp), Schmitt trigger and level shifter

ger and amplifier have the same basic configuration with a slight difference in the conductance ratio of the load transistors to ensure a positive resistance load in the amplifier and a negative resistance load in the Schmitt trigger. In addition, the Schmitt trigger has a level-shifting circuit

at the output (transistors $M_{11}$ to $M_{14}$) that is used to convert the Schmitt trigger output to digital CMOS levels. In the figure, $n$ is the ratio of the transconductance of the two load transistors ($n = \frac{W_3}{W_5} = \frac{W_4}{W_6}$) and is less than one for the amplifier. For the Schmitt trigger, $n$ is greater than one. $M_8$ is a short device that is used to latch the data in $\phi_{2d}$. The comparator and the DAC are designed as two independent circuits in order to test them separately. Three independent current steering cells with $0.1\ LSB$ variation in device current was included in the design to evaluate the comparator independently.

### *C. Logic for Reconfiguration*

Digital logic is used to control the DAC and comparator for DNL estimation. In a segmented DAC, the DNL due to the upper segment is dominant. Therefore, we have sorted the current sources based on the estimated DNL of the upper segment sources. DNL is estimated for every change in upper segment code. A $31 \times 3$ bit memory is used to store the DNL. The hardwired thermometer decoder of the upper segment is replaced with a magnitude comparator and a $5$ bit code-register. The magnitude comparator generates a 'HIGH' when the DAC input code is greater than or equal to code-register content. The sorting logic writes the new address of each current source according to the modified sequence. This uses an additional $31 \times 1$ bit memory. At reset, the code-register contains a default thermometer code. The block schematic of the reconfiguration scheme is shown in figure 8.

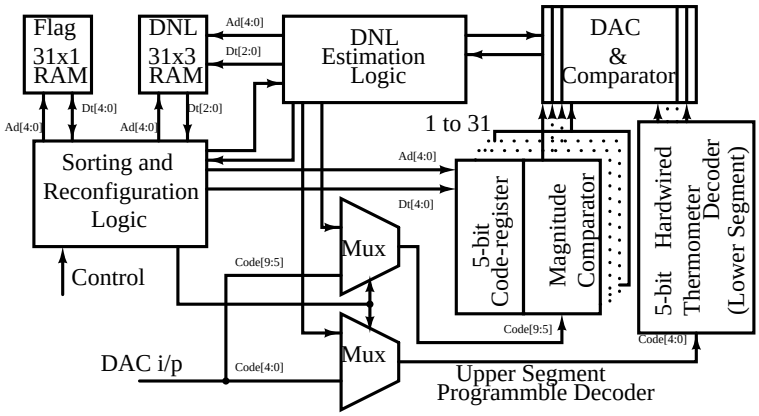


Fig. 8. Block schematic of the reconfiguration scheme.

## IX. Accuracy of the Circuits used for Measurement

The errors in the measurement arise due to errors in the reference current sources, comparator and mismatch in the load resistance. This is discussed in this section. Note that this technique cannot be used to estimate errors in the lower segment cells, since these errors are of the same order as the two extra cells used in the estimation.

### *A. Measurement Errors due to Resistor Mismatch*

Resistors at the DAC output convert the currents to voltages, which are the inputs to the comparator. The comparator always provides an output corresponding to difference of the two differential voltages. Therefore, the step size corresponding to two adjacent codes can be written as

$$V_{step} = I_{LSB}(R_1 + R_2) \tag{21}$$

where $I_{LSB}$ current difference between the two codes. From the above equation, it is clear that a mismatch in these resistors will create an error only if the two voltages being compared have a different common mode value. Since the output is fully differential, with every code change, the current is only switched from one arm to the other and the common mode voltage does not change. The only exception to this is the $0.5\ LSB$ cell, which has one arm connected to $V_{dd}$. Therefore, when the $0.5\ LSB$ cell is used, there is an error in the sampled input voltage. It is equal to $I_{0.5}\Delta R$, where $\Delta R$ is the difference between the two resistor values. In our case, this was estimated to be less that $0.001\ LSB$.

### *B. Measurement Errors due to the reference current sources*

The two extra cell (reference current) have the same accuracy as any lower segment current source. This introduces a measurement error. The worst case error occurs when both the current sources are used. In this case, the expected value of the error is $\sqrt{2}\ \sigma$. The lower segment current sources were sized so that the mismatch is $1.04\%$ (refer to Table I). Therefore the accuracy of these sources is 0.01 LSB. This was considered sufficient since the resolution of the measurement was $0.25LSB$.

### C. Amplifier and Latch errors

The comparator also has a residual offset which occurs due to the following:

1) Latch offset $V_{OSL}$: The input referred offset due to this $\frac{V_{OSL}}{A}$, where $A$ is the gain of the preamplifier. The gain cannot be made very large, since output offset storage technique is used to cancel out the preamplifier offset.
2) Charge injection and clock feed-through: Mismatch between the switches and capacitors, both at the input and output creates an additional offset. The residual offset due to this is given by:
$$\Delta V_q = (\frac{\Delta q_{in}}{C_{in}} + \frac{q_{in}}{C_{in}^2}\Delta C_{in}) + \frac{1}{A}(\frac{\Delta q_{out}}{C_{out}} + \frac{q_{out}}{C_{out}^2}\Delta C_{out}) \quad (22)$$
Here $q$, $C_{in}$ and $C_{out}$ are the nominal values of the charge injected by the switches (including clock feed-through effects), the sampling capacitors and the offset compensation capacitors. $\Delta C_{in}$ and $\Delta C_{out}$ occur due to random mismatch errors in the input and output capacitors. $\Delta q$, the mismatch in the charge injected by the switches is approximately given as:
$$\Delta q = \Delta C_{ox}(V_{gs} - V_T) + \Delta V_t C_{ox} + \Delta C_{ov} V_{DD} \quad (23)$$
Here $\Delta C_{ox}$, $\Delta V_t$ and $\Delta C_{ov}$ are due to random mismatches in the oxide capacitance, threshold voltages and overlap capacitances of the switches respectively.
3) Thermal noise: This occurs due to the sampled thermal noise due to the resistive elements at the input. It is of the order of $\sqrt{\frac{2KT}{C}}$ V.

Based on these considerations, the accuracy of the comparator was estimated to be $0.03LSB$. Most of this error is due to the sampled thermal noise.

### D. Expected worst case error

The total worst case error can be written as

$$V_{err} = I_{0.5}\Delta R + R\, I_{LSB}\sqrt{2}\,\sigma_{LSB} + \frac{V_{OSL}}{A} + \Delta V_q + \sqrt{\frac{2KT}{C}} \quad (24)$$
$$\approx R\, I_{LSB}\sqrt{2}\,\sigma_{LSB} + \sqrt{\frac{2KT}{C}} \quad (25)$$

In our design, this is about $0.05LSB$. Essentially, it is equivalent to measuring the output voltage of the DAC with an ADC that has 12 bit resolution and 13-14 bit accuracy.

## X. Performance, Area and Power overhead

Any extra load at the differential DAC output can degrade the performance of the DAC. The proposed scheme can influence both static and dynamic performance of the DAC. The half LSB current source introduces an offset of half LSB at DAC output. The offset due to the reverse saturation current of the switch is of the order of few tens of femto Amperes and can be neglected as compared to the LSB current of 1$\mu$A.

The capacitive load offered by switch is of the order of a few femto Farads. This is in parallel with output capacitance of the current steering cells which is of the order of pFs. This degrades the bandwidth by 0.1%. The asymmetric load offered by the half LSB source results in a output capacitance mismatch of about 0.05% (including the pad capacitance). This introduces an additional asymmetry in rise and fall time and hence there could be a change in the distortion characteristics of the DAC. However, simulations indicate that this change is negligible (change in the second harmonic distortion was less than 0.1%).

The analog circuits (comparator and the two LSB current steering cells) for DNL estimation requires about $0.06mm^2$. The magnitude comparators and the 5 bit code-register for all the 31 sources requires an additional area of $0.095mm^2$. If the sorting and reconfiguration logic is implemented on silicon it will occupy about $0.07mm^2$. A 30% area overhead is required to implement the estimation and reconfiguration circuit.

The two extra current sources consumes a continuous current 2$\mu$A. The comparator require about 20$\mu$A which can be turned of after DNL estimation. Therefore, during the test the circuit consumes about 70$\mu$W.

## XI. Results

Figure 9 shows the photograph of the die. A test board was developed to test the DAC as well as the reconfiguration scheme. It is also shown in figure 9. The DAC, the programmable decoder and the analog circuits for measurement were implemented on chip with core area of 0.83mm$^2$. The power consumption was about 10 mW at 78 MS/S and 3.3V.

### *A. Comparator performance*

As mentioned previously, three independent current steering cells with $0.1\ LSB$ variation in device current was included in the design to evaluate the comparator independently. This was

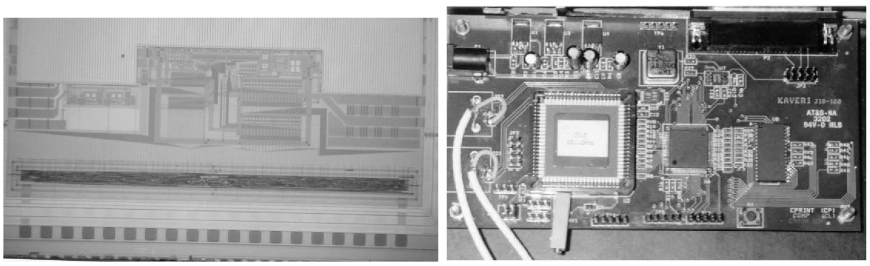

Fig. 9. Micro photograph of the die and Photograph of the test board

detected by the comparator. The comparator was further evaluated with the DAC in the test mode. Each of the upper segment cells was compared with every other cell. Figure 10 shows these test results for one of the samples. It is an indication of how well the upper segment sources are matched. In order to be absolutely certain that correct results were obtained, each comparison was repeated in the reverse direction i.e the results for $V_1 - V_2$ and $V_2 - V_1$ were obtained. This is shown in Figure 10. The boxes below the diagonal show the result of one comparison and the boxes above the diagonal show the reverse comparison. A white box indicates that the current through the cell indicated on vertical axis is larger than the one on the horizontal axis. A black box indicates the opposite result. Gray indicates that the two currents are equal to within the resolution of the comparator. Therefore, a box that is white above the diagonal should ideally be black below the diagonal. From the figure this is seen to be true. Along the diagonal all the boxes should be gray, since it is a comparison of a current source with itself. This is also seen in the figure.

These comparisons were made for ten samples. It was found that the comparator could detect a difference of $0.05LSB$.

### *B. DNL estimation and Reconfiguration*

The DAC was first characterized to study its DNL and INL. The DNL at critical points was estimated using the on-chip measurement scheme. The results were compared with external measurements. A sample result is shown in Figure 11. Clearly, a reasonable estimate of the DNL can be obtained.

The on-chip measurement data was used to optimize the switching sequence of the upper segment sources.This was done for seven samples. The DNL and INL before and after reconfiguration for all the seven samples are shown in figure 12. A significant improvement in INL was

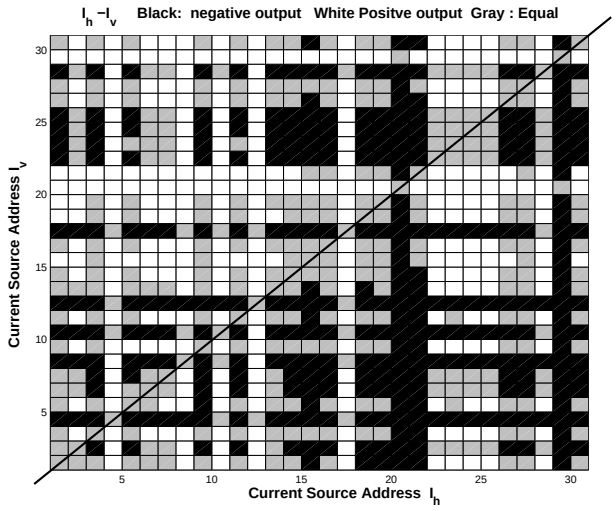


Fig. 10. Comparison of each cell current in upper segment with all other cell currents, a white box indicates that the current through the cell indicated on vertical axis is larger than the one on the horizontal axis. A black box indicates the opposite result.

observed in four of the seven samples(samples b,c,d,f) and no improvement was visible in three samples. Typically, there is no improvement if the INL before reconfiguration is already of the order of $|DNL|_{max}$. The maximum value of $|DNL|$ and $|INL|$ before and after reconfiguration is tabulated in table V. The experimental result shows that the average DNL is around 0.95 LSB and average INL after reconfiguration has decreased from 1.9 to to 1.15 LSB. In sample 'e' the INL has increased by 50% after reconfigutation. This is basically due to the inbalence the number of positive and negative DNLs. Similar result can be observed in the presents of large quadratic gradient mismatch.

## XII. Conclusions

The reconfiguration scheme presented here can be used to get improved linearity and hence a higher INL yield in segmented current steering DACs. The lowest INL that can be obtained is the same as the maximum DNL. The trade off here is the total area of the current sources versus the area of the measuring circuits. It is particularly useful in segmented DACs, where there is significant DNL when there is a change over from the lower to the upper segment. At the same time, the INL is contributed mostly by the upper segment sources and errors in these sources can be measured with sufficient resolution using a reasonable amount of area. Further improvements in the measuring technique is needed to get closer to the simulation results.

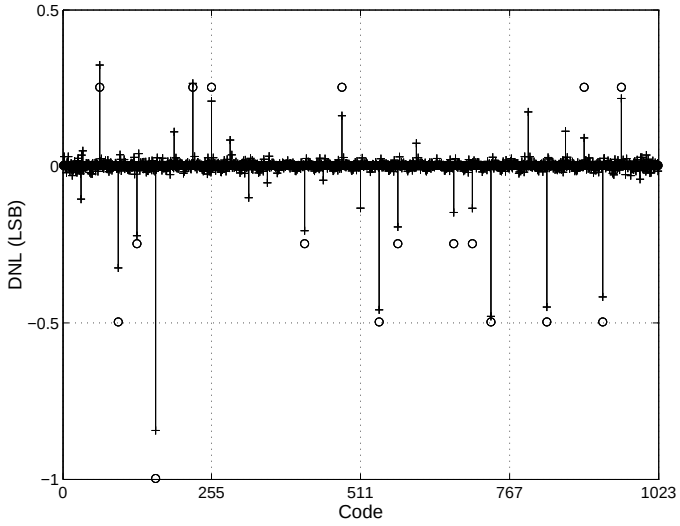


Fig. 11. DNL from external measurement and on-chip estimation. Continuous line with cross indicate DNL from external measurement, circle indicate DNL from on-chip estimation.

## REFERENCES

[1] M.J.M. Pelgrom, A.C.J. Duimaijer, and A.P.G. Welbers, "Matching Properties of MOS Transistors," *IEEE Journal of Solid State Circuits*, vol. 24, no. 5, pp. 1433–1439, October 1989.

[2] A. Van den Bosch, M. A.F. Borremans, M. S.J. Steyaert, and W. Sansen, "A 10-bit 1-GSamples/s Nyquist Current-Steering CMOS D/A Converter," *IEEE Journal of Solid State Circuit*, vol. 36, no. 3, pp. 315–324, March 2001.

[3] C.H Lin, and K. Bult, "A 10-b 500MSamples/s CMOS DAC in 0.6 $mm^2$," *IEEE Journal of Solid State Circuits*, vol. 33, no. 12, pp. 1948–1958, December 1998.

[4] G. A.M. Van der Plas, J. Vandenbussche, W. Sansen, M. S.J. Steyaert, and G. G.E. Gielen, "A 14-bit Intrinsic Accuracy $Q^2$ Random Walk CMOS DAC," *IEEE Journal of Solid State Circuits*, vol. 34, no. 12, pp. 1708–1718, December 1999.

[5] J.A. Starzyk, R.P. Mohn, and L. Jing, "A cost-Effective Approach to Design and Layout of a 14-b Current-Steering DAC Macrocell," *IEEE Transactions on Circuits and Systems- I Regular Papers*, vol. 51, no. 1, pp. 196–200, January 2004.

[6] J. M.Steyaert, G.Gielen, and W. Sansen, "A Gradient-Error and Edge Effect Tolerant Switching Scheme for a High Accuracy DAC ," *IEEE Transactions on Circuits and Systems- I Regular Papers*, vol. 51, no. 1, pp. 191–195, January 2004.

[7] Y. Cong, and R. L. Geiger, "Switching Sequence Optimization for Gadient Error Compensation in Thermometer-Decoder DAC Arrays," *IEEE Transactions on Circuits and Systems- II Analog and Digital Signal Processing*, vol. 47, no. 7, pp. 585–595, July 2000.

[8] Yonghua Cong, and Randall.L.Gaiger, "A 1.5-V 14-Bit 100-MS/s Self-Calibrated DAC," *IEEE Journal of Solid State Circuits*, vol. 38, no. 12, pp. 2051–2060, December 2003.

[9] J. A. M. Marques, M. S.J.Steyaert, and W. M. Sansen, "A 12-Bit Intrinsic Accuracy High-Speed CMOS DAC," *IEEE Journal of Solid State Circuit*, vol. 33, no. 12, pp. 1959–1969, December 1998.

[10] Y. Nakamura, T. Miki, A. Maeda, H. Kondoh, and N. Yazawa, "A 10-b 70-MS/s CMOS D/A Converter," *IEEE Journal of Solid State Circuits*, vol. 26, no. 4, pp. 637–642, April 1991.

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

| Chip | $\|DNL\|_{max}$ | $\|INL\|_{max}$ before | $\|INL\|_{max}$ after |
|---|---|---|---|
| a | 0.82 | 0.895 | 1 |
| b | 0.95 | 2.48 | 0.9 |
| c | 0.83 | 1.475 | 0.8 |
| d | 1.25 | 2.9 | 1.45 |
| e | 1.18 | 1.2 | 1.7 |
| f | 1.15 | 3.3 | 1.2 |
| g | 0.476 | 1.05 | 1.1 |

TABLE V

$|DNL|_{max}$ AND $|INL|_{max}$ BEFORE AND AFTER RECONFIGURATION

[12] Yonghua Cong, and Randall.L.Gaiger, "Formulation of inl and dnl yield estimation in current-steering d/a converters," in *IEEE International Symposium on Circuits and Systems(ISCAS2002)*, vol. 3, 2002, pp. 149–152.

[13] A.Van den Bosch, M.Steyaert, and W. Sansen, "An accurate statistical yield model for CMOS current-steering D/A converters," in *IEEE International Symposium on Circuits and Systems(ISCAS2000)*, vol. 4, 2000, pp. 105–108.

[14] W.H.Beyer, Ed., *CRC Handbook of Tables for Probability and Statistics*. CRC Press, 1968.

[15] K.P.S. Rafeeque, "Area Efficient Current Steering DAC Using Current Tuning," in *IEEE Asia-Pacific Conference on Circuits and Systems(APCCAS2002)*, vol. 1, 2002, pp. 559–564.

[16] B. Razavi, *Principles of Data Conversion System Design*. Wiley-IEEE Press, 2001.

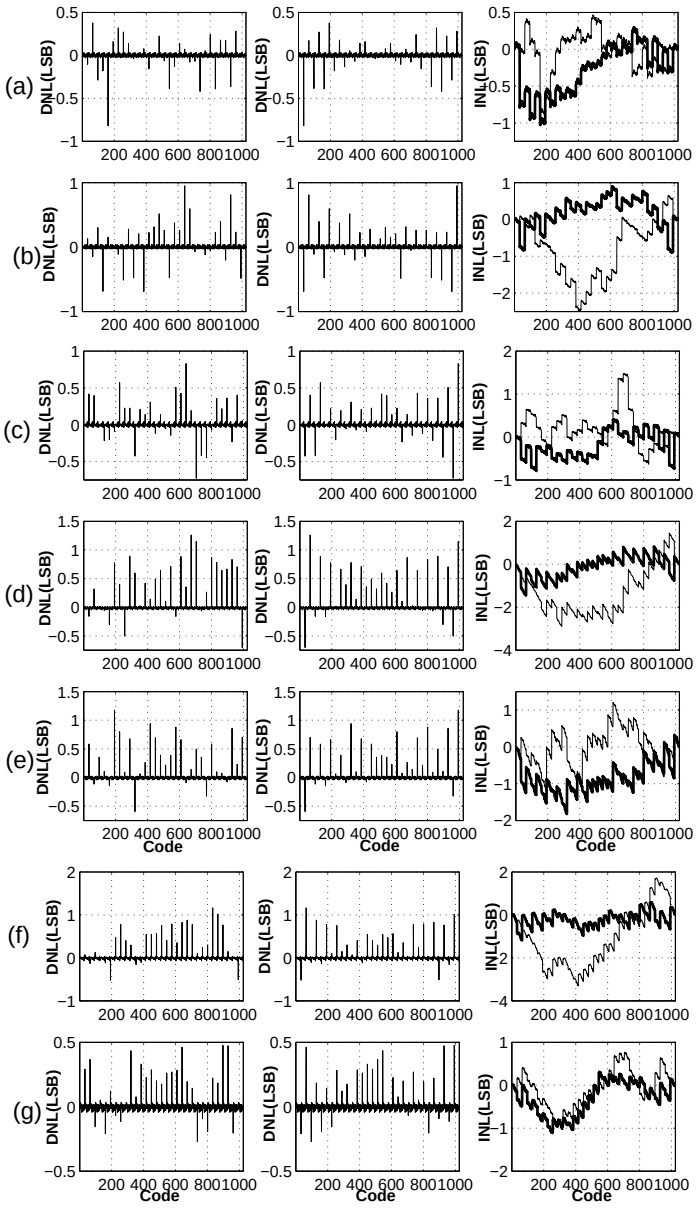


Fig. 12. DNL and INL before and after reconfiguration for seven samples. For each sample first column indicate DNL before reconfiguration, middle column indicate DNL after reconfiguration and INL before and after(thick line) reconfiguration in the last column respectively.