1-D and 2-D switching strategies achieving near optimal INL for thermometer-coded current steering DACs

Zhongjun Yu, Degang Chen & Randy Geiger

Iowa State University
Ames, Iowa 50011, USA

Abstract

It is known that thermometer coded current steering DACs have very small DNLs but much larger INLs, even with the best switching strategies available in the literature. In this paper, we present algorithms for generating near optimal 1-D switching strategies for any types of gradient errors and 2-D switching strategies for linear gradient errors. In both cases, the new switching strategies result in total INL that is less than the DNL. This is far less than what was achievable by existing switching strategies and is very close to the absolute lower limit of ½DNL.

1. Introduction

In an n-bit current-steering DAC, an array of $N = 2^n - 1$ current sources (excluding a dummy) are used, each providing an intended current I called 1 LSB (Lease Significant Bit). When the digital input is D ($0 \le D \le N$), D current sources are steered to an external resistor, R (typically R = 50 Ω), generating a voltage output, $V_o = I \cdot R \cdot D$. However, due to process and temperature variations, mismatch always exits between the current sources, causing nonlinear errors in the output voltage. The performance of current-steering DACs is generally characterized by their static and dynamic nonlinearities [1-3]. Static nonlinearities characterize DAC performance at dc or low frequency. Static nonlinearity is typically measured by integral and differential nonlinearities (INL and DNL) as expressed by:

$$\mathrm{INL(D)} = \frac{\mathrm{I(D)} - \mathrm{I(0)}}{[\mathrm{I(N)} - \mathrm{I(0)}]/\mathrm{N}} - \mathrm{D} = \sum_{i=0}^{D} \varepsilon_i$$

$$\mathrm{DNL(D)} = \frac{\mathrm{I(D)} - \mathrm{I(D-1)}}{[\mathrm{I(N)} - \mathrm{I(0)}]/\mathrm{N}} - 1 = \varepsilon_D$$

where I(D) is the actual analog output in units of LSB, and I(0) is the offset current. ε_i is the relative deviation of the i-th current source from the current sources' average value. In addition to random errors (which are reduced by increasing area), significant mismatch between current sources can be caused by gradient errors. Many factors may cause gradient over a current source array. For example, the spread of doping and oxide thickness over the wafer or the voltage drop along the power line can cause approximately linear gradient errors [6-8]. Temperature gradients and die stress may introduce approximately quadratic gradient errors. The overall gradient error distribution is given by superimposing these error components. The gradient errors in current arrays can become very significant and introduce large systematic errors. Decoding schemes can be used to improve both static and dynamic performance significantly. Two decoding schemes are generally used: binary-weighted and thermometer-decoded. The advantage of a binary-weighted DAC is its simplicity and low power consumption, since no decoding logic is required [4, 5]. Drawbacks include possible non-monotonicity in the transfer characteristic and potentially large DNL at major carrier transitions. In contrast, monotonicity is guaranteed in a thermometer decoded DAC. However, additional circuitry is required to implement the binary-to-thermometer decoder. In addition to monotonicity, another major advantage of thermometer decoded DACs is that their DNL is much smaller than that of the binary-weighted DAC. Unfortunately, the INL for both schemes is at the same level in general if no special switching strategy is implemented.

In this paper, we present a new switching strategy for thermometer decoded DAC that achieves INL that is less than DNL! This is far less than what was achievable by existing switching strategies and is very close to the absolute lower limit of ½DNL that is only theoretically possible.

2. Existing switching strategies

As pointed out in last section, in general, thermometer coding and binary-weighted coding have the same INL, especially so when the errors in the current sources are due to random errors. However, if the errors are due to gradient errors, various switching schemes can be used to decrease the INL.

The well-known row-column switching scheme [4, 6, 7] reduces the switching optimization problem to a one-dimensional space. In this scheme, the spatial gradient is averaged in two directions and the sequences for row and column selection are optimized independently. Furthermore, complexities in decoder design and layout are significantly reduced due to the column and row separation. However, the row-column switching schemes are inherently insufficient for two-dimensional gradient error compensation. The "Q2 Random Walk" scheme [8] divides the current sources into regions and has two-step hierarchical switching scheme. The switching for each region can be used to compensate for quadratic errors and the switching within each region is for linear error compensation. This hierarchical switching scheme allows optimization in 2-D space with penalty of very complex routing. INL bounded switching scheme [9] achieves optimal performance for one-dimensional array with linear gradients. However, if the current sources are not arranged in a 1-D array or if errors in the 1-D array are not caused by linear gradient, it is not optimal and it needs a searching algorithm, whose computational complexity is prohibitively high in the worst case.

3. A new 1-D switching strategy

The performance of the 1-D switching strategy in [9] is excellent. However, except for the simple situation of 1-D linear gradient, the searching algorithm may require large computational time. Even for linear gradient over the die, the mismatch errors in a 2-D current source array are actually nonlinear when they are sorted into a 1-D vector. For such cases, because each step of the algorithm involves searching, the worst case complexity is in the order of $(N/2)^N$, where N = 2^n, which quickly becomes an astronomical number even for a modest n.

0-7803-7761-3/03/$17.00 ©2003 IEEE

The proposed algorithm achieves the same INL performance for any nonlinear 1-Darray as the algorithm in [9] for linear, but it has a complexity of only $N\log(N)$! We achieve this by eliminating the searching in each step. The basic idea of the proposed algorithm is as follows. First we sort the current sources by their corresponding error ε_i's in ascending order and partition them into positive, zero, and negative groups. Then we start by switching in a current source having an error close to –½DNL. In the following steps, each time, we try to turn on the current source with the smallest (in magnitude) negative error available in order to make INL(k) approach or just pass –½DNL, which is the absolute optimal INL obtainable by any switching strategy. When this happens, we pick the current source which has the most positive error available and then try to turn on current sources with the smallest positive ε available to make INL(k) approach or just pass +½DNL. When this happens, we pick the current source with the most negative error. We continue these steps until all the remaining current sources have been picked. Notice that in each step, we don't need to search, we just pick the current source from the sorted groups sequentially. The pseudo-code of the algorithm is described as follows with the help of the Figure 1.

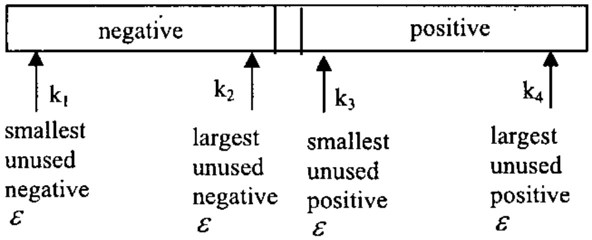

negative
positive
k_1
k_2
k_3
k_4
smallest unused negative ε
largest unused negative ε
smallest unused positive ε
largest unused positive ε

Figure 1: Illustration of 1-D algorithm

Pseudo-code of the proposed 1-D algorithm:

```
Sort $\varepsilon$'s into three vectors as shown in Figure 1
Initialize k=1;
Step k: $DNL(k)=\varepsilon_{neg-mid}$, $INL(k)=\varepsilon_{neg-mid}$;
Step (k+1):
    Case 1: $INL(k)<0$,
            if $INL(k)+\varepsilon_{k2} \geq -DNL/2$
    (#)             pick $\varepsilon_{k2}$, and let $DNL(k+1)=\varepsilon_{k2}$,
                    $INL(k+1)=INL(k)+\varepsilon_{k2}$,
                    $k_2=k_2-1$; $k=k+1$,
                    return to step (k+1);
            else
                    if $|INL(k)+\varepsilon_{k2}|>|INL(k)+\varepsilon_{k4}|$
                        pick $\varepsilon_{k4}$, and let $DNL(k+1)=\varepsilon_{k4}$,
                        $INL(k+1)=INL(k)+\varepsilon_{k4}$,
                        $k_4=k_4-1$; $k=k+1$,
                        return to step (k+1);
                    else    go to (#);
    Case 2: INL(k)>0, it's parallel to the above with proper
            modification;
    Case 3: INL(k)=0, go to step k;
Final step: sequentially pick all the points in the zero group.
```

Note 1: In real situations, the zero error group will be empty (i.e. no current source has exactly zero error) and the final step will not be used. Similarly case 3 of step k+1 will not be activated in real applications.

Note 2: The above is only the pseudo-code. In the real code, mechanisms need to be provided to prevent k_1 from crossing k_2, and prevent k_3 from crossing k_4.

Note 3: When picking the very first point (i.e. in step k), a search can be used to ensure that INL(1) ≥ –½DNL, but as close as possible to –½DNL. Since this is one-time search, it does not add much complexity.

Proposition The INL in the above algorithm will be bounded by DNL, namely, $DNL/2 \leq INL \leq DNL$

Proof: First, the left half of the inequality is true for any switching strategy. To prove this, suppose $|\varepsilon(k_0)|=DNL$ for some k_0. Then

$$\begin{aligned} \varepsilon(k_0) &= INL(k_0)-INL(k_0-1) \\ DNL &= |\varepsilon(k_0)| = |INL(k_0)-INL(k_0-1)| \\ &\leq |INL(k_0)|+|INL(k_0-1)| \\ &\leq \max_k |INL(k)| + \max_k |INL(k)| = 2INL \\ &\therefore\ INL \geq DNL/2 \end{aligned} \tag{1}$$

Now let's prove by induction that our algorithm achieves $INL \leq DNL$. To prove $INL \leq DNL$, it suffices to prove $INL(k) \leq DNL$ for all k. The initial step is clearly satisfied since ε_1 is the mid-point of the vector of negative current source errors, $\therefore |INL(1)|=|\varepsilon_1| \leq DNL$. For the induction step, let's assume $|INL(k)| \leq DNL$, and prove $|INL(k+1)| \leq DNL$ by contradiction. There are 3 cases in the algorithm. In case 3, $INL(k+1)=INL(k)+\varepsilon_{mid-neg} = 0+\varepsilon_{mid-neg}$

$$\therefore\ |INL(k+1)| = \varepsilon_{mid-neg} \leq DNL \tag{2}$$

Case 2 is exactly symmetric to case 1 and its proof is the same as case 1. Let's prove case 1 here. $\because -DNL \leq INL(k) < 0$, $\therefore$ there exists at least one unused $\varepsilon_{k4}>0$ since otherwise the rest unused ε's will be all negative, leading to $INL(k)+\sum_{unused}\varepsilon<0$, namely, $\sum_1^N \varepsilon < 0$, contradicting to $\sum_1^N \varepsilon = 0$.

In the (k+1)th step, if $INL(k+1)=INL(k)+\varepsilon_{k4}$ is used, then

$$\begin{aligned} |INL(k+1)| &= |INL(k)+\varepsilon_{k4}| \leq \big|\, \varepsilon_{k4} - |INL(k)| \,\big| \\ &\leq \max\{|\varepsilon_{k4}|, |INL(k)|\} \leq DNL \end{aligned} \tag{3}$$

If $INL(k+1)=INL(k)+\varepsilon_{k2}$ is used, there are two possibilities:

a) $INL(k)+\varepsilon_{k2} \geq -DNL/2$ or

b) $INL(k)+\varepsilon_{k2} < -DNL/2$, but $|NL(k)+\varepsilon_{k2}| \leq |INL(k)+\varepsilon_{k4}|$

In case a, $INL(k+1)=INL(k)+\varepsilon_{k2} \geq -DNL/2$ but $INL(k)+\varepsilon_{k2}<0$

$$\begin{aligned} &\therefore\ 0 > INL(k+1) \geq -DNL/2 \\ &\therefore\ |INL(k+1)| \leq DNL/2 < DNL \end{aligned} \tag{4}$$

In case b, $|INL(k+1)|=|INL(k)+\varepsilon_{k2}| \le |INL(k)+\varepsilon_{k4}|$

$$= |\varepsilon_{k4}-|INL(k)|| \le \max\{\varepsilon_4, |INL(k)|\} \le DNL \qquad (5)$$

$$\therefore\ |INL(k+1)| \le DNL \quad \text{for case 1} \qquad (6)$$

From (2) and (6), we proved $|INL(k+1)| \le DNL$ for all cases.

By induction, $|INL(k)| \le DNL$ for all k. Hence,

$$\therefore\ INL \le DNL \qquad (7)$$

$$\therefore\ \text{From (1) and (7), } \tfrac{1}{2}DNL \le INL \le DNL$$

Note: in the algorithm, we push $|INL(k)|$ to be very close to ½DNL, therefore most likely, the INL in the above algorithm is very close to ½DNL.

1-D Simulation Results

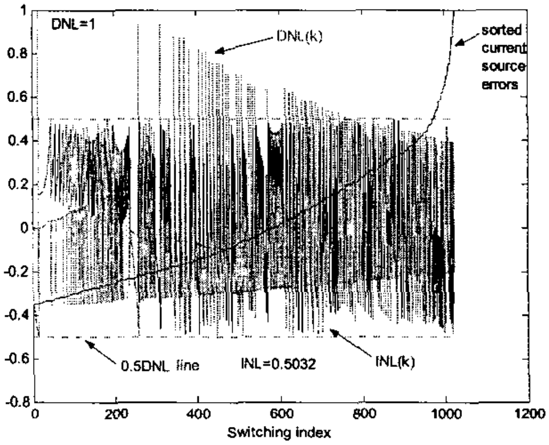


Figure 2. Simulation results for 1-D algorithm

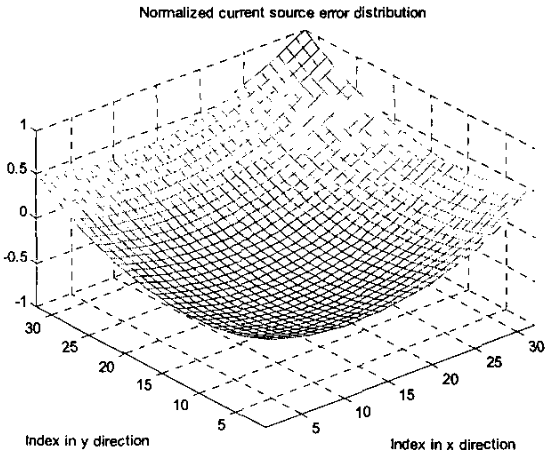


Figure 3. Normalized gradient error distribution

Figure 2 gives the simulation result for a 10-bit DAC having both linear gradient error and quadratic gradient error. The gradient error for a current source at location (x, y) can be represented by:

$$\varepsilon(x,y)=\varepsilon_l(x,y)+\varepsilon_q(x,y)$$

where $\varepsilon_l(x,y) = g_l x\cos\theta + g_l y\sin\theta$ is the linear gradient error, g_l and θ are the strength and angle of the linear gradient, $\varepsilon_q(x,y) = g_q(x^2+y^2)-m_0$ is the quadratic gradient error. g_q is determined dominantly by the die bonding techniques and m_0 is the average quadratic error of the current sources. For an MxM current source matrix, the errors are normalized so that the maximum error magnitude will be equal to 1. $\varepsilon_n(x,y)$ can be expressed as:

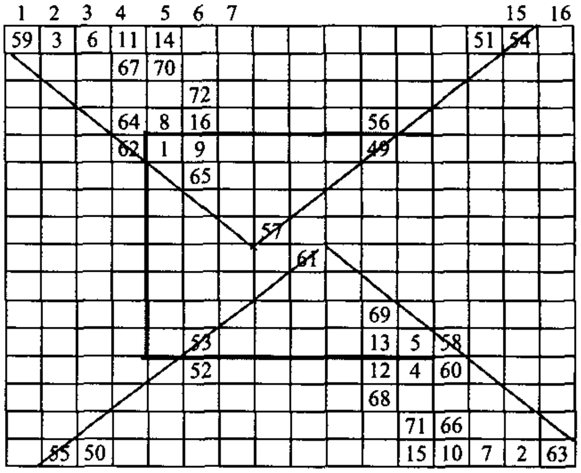
$$\varepsilon_n(x,y) = \varepsilon(x,y)/S$$

where S is the scaling factor and is equal to $[((M-1)/2)*(\sqrt{2}g_l+2g_q-m_0]$.

The normalized gradient error for 10-bit DAC is shown in Figure 3. The ratio of linear gradient error to quadratic error is 1. Figure 2 shows us that, by using the proposed 1-D algorithm, the normalized DNL is 1 and the normalized INL is 0.5032, which is only slightly > ½ DNL, which is the absolute theoretical limit.

4. A 2-D switching strategy

The proposed switching strategy in section 3 can provide an overall INL only slightly above the absolute lower bound of INL, i.e. ½ of the largest current source error. However, the algorithm needs to know the angle of the linear gradient, and, if nonzero, the quadratic gradient relative strength. Such information is typically not available easily or it could change from die to die. Therefore, we need an algorithm that can generate systematically switching strategies for various size current source arrays and requires information only on geometric location but not the actual gradient errors. In this section, we introduce such an algorithm.



1	2	3	4	5	6	7								15	16
59	3	6	11	14									51	54	
			67	70											
					72										
			64	8	16					56					
			62	1	9					49					
					65										
							57								
								61							
										69					
					53					13	5	58			
					52					12	4	60			
										68					
											71	66			
	55	50									15	10	7	2	63

Figure 4 Illustration of a 8-bit DAC switching sequence

Due to space limitation we will only give a very brief description of the basic concept of the algorithm. Consider an 8-bit DAC. The current sources will be a 16X16 array. We divide this array into two parts: the central 8X8 array and the outside part. In order to have INL close to ½ DNL, we start with current source 1 whose error is ½DNL. To keep INL close to ½DNL, current source 2 at the far opposite side is picked followed by its mirror image source 3. The INL after this step would be ½ DNL. So we choose current source 4 so that the INL for this step would be about 0. Now we choose the current sources 5 through 8 in the mirror image pattern of 1 through 4. At the end of this 8-source pattern, INL(8) is exactly 0. Repeat this 8 sources pattern but each time right shift the starting source one space, until the second last current source 49 in the row. The sources along the diagonal are then picked as indicated by 57 through 64. Then we repeat this process until all rows in the up-down triangles are used, with the first 8-pattern as indicated by 65 through 72. The left-right triangles are dealt with in the exact fashion but with the roles of column and row switched.

2-D Simulation Results

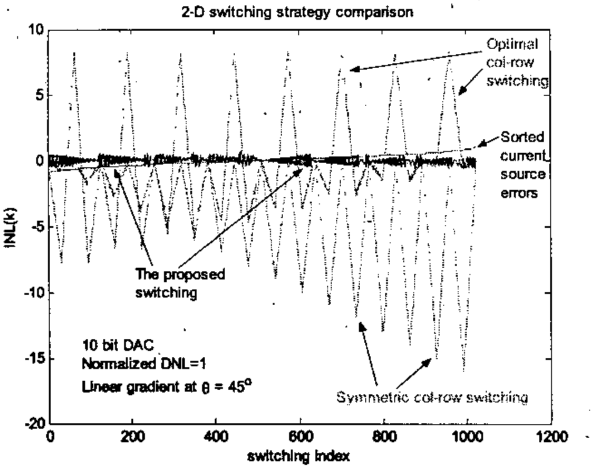


Figure 5. 2-D switching strategy comparison

A 64x64 current source array for a 12-bit and a 32x32 current source array for a 10-bit thermometer decoded DAC are used to compare the new 2-D switching strategy against the conventional sequences. The simulation results are shown in Figures 5 and 6. In the simulation, the error distributions across the matrix are normalized so that the DNL is always 1 and INL is measured in terms of DNL. In Figure 5, the angle of the linear gradient is 45°. The INL for both optimal column-row switching [9] and symmetric column-row switching are about 8 DNL and 16 DNL respectively, which is far beyond the absolute optimal INL. The INL from the proposed switching strategy is about 0.8 DNL, which is very close to the absolute optimal INL of 0.5 DNL. Furthermore, we simulated the 12-bit DAC with the angle of the linear gradient varying from 0° to 360°. The simulation results are summarized in Figure 6. Notice that the INL from the proposed 2-D switching algorithm is always around 0.7-0.8 DNL, regardless of the gradient angles. In contrast, the performance of the other two switching strategies varies significantly with the gradient angles.

5. Conclusion

In this paper, an algorithm is introduced for generating near optimal switching sequences for one-dimensional DAC current arrays. The switching sequence results in the DAC's INL about ½ the maximum relative deviation of the current sources from their average current value. The algorithm also provides near optimal switching sequences for any types of gradient error conditions. A 2-D switching algorithm also presented which significantly reduces the linearity error due to gradient mismatch, compared with what is obtainable with the best published switching strategy.

6. References

[1] J. Wikner and N. Tan, "Modeling of CMOS Digital-to-Analog Converters for Telecommunication," *IEEE Trans. Circuits and Systems II,* vol.46, pp.489-499, May, 1999.

[2] D. Johns and K. Martin, *Analog Integrated Circuit Design*, Wiley, New York, 1997.

[3] B. Razavi, *Principles of Data Conversion System Design*, IEEE Press, Piscataway, NJ, 1995.

[4] C.-H. Lin and K. Bult, "A 10b, 500MSample/s DAC in 0.6 mm^{2}", *IEEE J. Solid-State Circuits*, vol.33, pp.1948-1958, Dec. 1998.

[5] Bastos, M. Steyaert, and W. Sansen, "A High Yield 12-bit 250-MS/s CMOS D/A Converter," *Proc. IEEE 1996 Custom Integrated Circuits Conf.*, pp. 431-434, May 1996.

[6] Y. Nakamura, T. Miki, A. Maeda, H. Kondoh, and N. Yazawa, "A 10-bit 70MS/s CMOS D/A Converter," *IEEE. J. Solid-state Circuits*, vol. 26, pp. 637-642, Apr. 1991.

[7] J. Bastos, A. Marques, M. Steyaert, and W. Sansen, "A 12 bit Intrinsic accuracy High Speed CMOS DAC", *IEEE J. Solid-State Circuits*, vol.33, pp.1959-1969, Dec. 1998.

[8] G. Van der Plas, J. Vandenbussche, W. Sansen, M. Steyaert, and G. Gielen, "A 14-bit intrinsic accuracy Q^{2} Random Walk CMOS DAC," *IEEE J. Solid-State Circuits*, vol. 34, pp. 1708-11718, Dec. 1999.

[9] Y. Cong and R.Geiger, "Switching Sequence Optimization for Gradient error Compensation in Thermometer-Decoded DAC Arrays", *IEEE Trans. Circuits and Systems II,* vol.47, pp.585-595, July 2000.

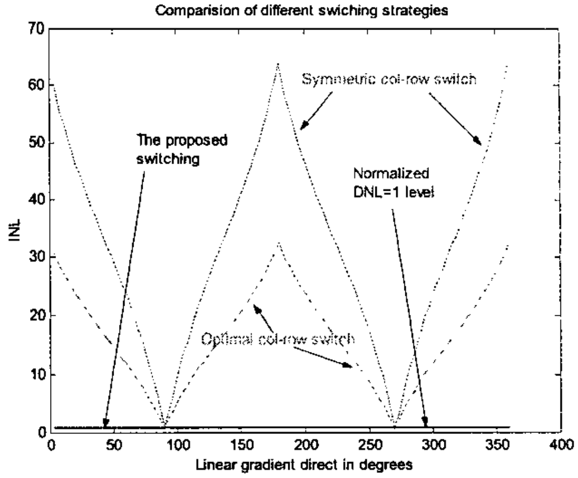


Figure 6. Comparison of different switching strategies