TABLE II
COMPARISON OF RECEIVER FRONTENDS

	This work	[7]	[8]*	[9]*	[10]	[11]
CMOS technology	0.18 µm	0.18 µm	90 nm	90 nm	0.18 µm	90 nm
RF freq. (GHz)	5.2	0.9	2.4	2.4	5.8	2.4
V_{DD} (V)	0.6	0.5	0.6	0.6	1.5	1.2
DC power (mW)	2.1	3.2	4	3.6	17.25	3.6
Gain (dB)	14.5	12	—	—	15.7	—
IIP3 (dBm)	-16	-14	-10.5	-12.5	-20.56	-13
DSB noise figure (dB)	8	9	16	9	10.6	10

*Baseband circuits were also integrated in these works.

5-GHz RF output at a LO power level of 6 dBm. By fixing the IF frequency, the conversion gain versus the LO frequency was characterized. The measured RF power versus IF power indicates an input-referred 1-dB compression point $(P_{\mathrm{in}-1\ \mathrm{dB}})$ of -16 dBm and a saturated output power (P_{sat}) of -1 dBm.

The performance of the receiver and transmitter frontends is summarized in Table I. According to the experimental results, the proposed circuit topologies demonstrate the potential of implementing CMOS RF frontends for ultra-low-power and ultra-low-voltage applications at multi-gigahertz frequencies. A comparison with other reported low-voltage receiver front-ends is tabulated in Table II.

V. CONCLUSION

Using a standard 0.18-μm CMOS process, fully integrated transmitter and receiver frontends are implemented at the 5-GHz frequency band. With the proposed design techniques, the fabricated RF frontends are able to operate at a reduced supply voltage of 0.6 V with ultra-low power consumption while maintaining reasonable circuit performance in terms of gain, linearity, and noise figure for short-range wireless communications.

ACKNOWLEDGMENT

The authors would like to thank National Chip Implementation Center (CIC), Hsinchu, Taiwan, for chip fabrication and National Nano Device Laboratories (NDL), Hsinchu, Taiwan, for chip measurement.

REFERENCES

[1] N. Stanic, P. Kinget, and Y. Tsividis, “A 0.5 V 900 MHz CMOS receiver front end,” in *IEEE Symp. VLSI Circuits Dig. Techn. Papers*, Jun. 2006, pp. 228–229.

[2] M. N. El-Gamal, K. H. Lee, and T. K. Tsang, “Very low-voltage (0.8 V) CMOS receiver frontend for 5 GHz RF applications,” *IEE Proc. Circuits, Devices Syst.*, vol. 149, no. 5/6, pp. 355–362, Oct.–Dec. 2002.

[3] M. Harada, T. Tsukahara, J. Kodate, A. Yamagishi, and J. Yamada, “2-GHz RF front-end circuits in CMOS/SIMOX operating at an extremely low voltage of 0.5 V,” *IEEE J. Solid-State Circuits*, vol. 35, no. 12, pp. 2000–2004, Dec. 2000.

[4] A.-S. Porret, T. Melly, C. Christian, and E. A. Vittoz, “A low-power low-voltage transceiver architecture suitable for wireless distributed sensors network,” in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2000, vol. 1, pp. 56–59.

[5] H.-H. Hsieh and L.-H. Lu, “Design of ultra-low-voltage RF frontends with complementary current-reused architectures,” *IEEE Trans. Microw. Theory Tech.*, vol. 55, no. 7, pp. 1445–1458, Jul. 2007.

[6] B. Kim, J.-S. Ko, and K. Lee, “A new linearization technique for MOSFET RF amplifier using multiple gated transistors,” *IEEE Microw. Guided Wave Lett.*, vol. 10, no. 9, pp. 371–373, Sep. 2000.

[7] N. Stanić, P. Kinget, and Y. Tsividis, “A 0.5 V 900 MHz CMOS receiver front end,” in *IEEE VLSI Circuits Symp. Tech. Dig.*, Jun. 2006, pp. 228–229.

[8] A. Balankutty, S.-A. Yu, Y. Feng, and P. Kinget, “A 0.6 V 32.5 mW highly integrated receiver for 2.4 GHz ISM-band applications,” in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2008, pp. 366–367.

[9] A. Liscidini, M. Tedeschi, and R. Castello, “A 2.4 GHz 3.6 mW 0.35 mm^2 quadrature front-end RX for ZigBee and WPAN applications,” in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2008, pp. 370–371.

[10] X. Wang and R. Weber, “A novel low power low voltage LNA and mixer for WLAN IEEE 802.11a standard,” in *Proc. IEEE Topical Meet. SiRF*, Sep. 2004, pp. 231–234.

[11] M. Tedeschi, A. Liscidini, and R. Castello, “A 0.23 mm^2 free coil ZigBee receiver based on a bond-wire self-oscillating mixer,” in *Proc. ESSCIRC*, Sep. 2008, pp. 430–433.

High Throughput DA-Based DCT With High Accuracy Error-Compensated Adder Tree

Yuan-Ho Chen, Tsin-Yuan Chang, and Chung-Yi Li

***Abstract*—In this brief, by operating the shifting and addition in parallel, an error-compensated adder-tree (ECAT) is proposed to deal with the truncation errors and to achieve low-error and high-throughput discrete cosine transform (DCT) design. Instead of the 12 bits used in previous works, 9-bit distributed arithmetic-precision is chosen for this work so as to meet peak-signal-to-noise-ratio (PSNR) requirements. Thus, an area-efficient DCT core is implemented to achieve 1 Gpels/s throughput rate with gate counts of 22.2 K for the PSNR requirements outlined in the previous works.**

***Index Terms*—Distributed arithmetic (DA)-based, error-compensated adder-tree (ECAT), 2-D discrete cosine transform (DCT).**

I. INTRODUCTION

Discrete cosine transform (DCT) is a widely used tool in image and video compression applications [1]. Recently, the high-throughput DCT designs have been adopted to fit the requirements of real-time applications [2]–[11].

The multiplier-based DCTs were presented and implemented in [2] and [3]. To reduce area, ROM-based distributed arithmetic (DA) was applied in DCT cores [4]–[6]. Uramoto *et al.* [4] implemented the DA-based multipliers using ROMs to produce partial products together with adders that accumulated these partial products. In this way, instead of multipliers, the DA-based ROM can be applied in a DCT core design to reduce the area required. In addition, the symmetrical properties of the DCT transform and parallel DA architecture can be used in reducing

Manuscript received June 29, 2009; revised September 17, 2009 and November 05, 2009. First published January 12, 2010; current version published March 23, 2011. This work was supported in part by the National Science Council under Project NSC 98-2221-E-007-095.

The authors are with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: yhchen@larc.ee.nthu.edu.tw; yhchen@yard.ee.nthu.edu.tw; tyc@ee.nthu.edu.tw; cyli@larc.ee.nthu.edu.tw).

Digital Object Identifier 10.1109/TVLSI.2009.2037968

1063-8210/$26.00 © 2010 IEEE

the ROM size in [5] and [6], respectively. Recently, ROM-free DA architectures were presented [7]–[11]. Shams *et al.* employed a bit-level sharing scheme to construct the adder-based butterfly matrix called new DA (NEDA) [7]. Being compressed, the butterfly-adder-matrix in [7] utilized 35 adders and 8 shift-addition elements to replace the ROM. Based on NEDA architecture, the recursive form and arithmetic logic unit (ALU) were applied in DCT design to reduce area cost [8], [9]. Hence the NEDA architecture is the smallest architecture for DA-based DCT core designs, but speed limitations exist in the operations of serial shifting and addition after the DA-computation. The high-throughput shift-adder-tree (SAT) and adder-tree (AT), those unroll the number of shifting and addition words in parallel for DA-based computation, were introduced in [10] and [11], respectively. However, a large truncation error occurred. In order to reduce the truncation error effect, several error compensation bias methods have been presented [12]–[14] based on statistical analysis of the relationship between partial products and multiplier-multiplicand. However, the elements of the truncation part outlined in this work are independent so that the previously described compensation methods cannot be applied.

This brief addresses a DA-based DCT core with an error-compensated adder-tree (ECAT). The proposed ECAT operates shifting and addition in parallel by unrolling all the words required to be computed. Furthermore, the error-compensated circuit alleviates the truncation error for high accuracy design. Based on low-error ECAT, the DA-precision in this work is chosen to be 9 bits instead of the traditional 12 bits so as to achieve the peak-signal-to-noise-ratio (PSNR) [1] requirements. Therefore, the hardware cost is reduced, and the speed is improved using the proposed ECAT.

This brief is organized as follows. In Section II, the mathematical derivation of the distributed arithmetic is given. The proposed ECAT architecture is discussed in Section III. The proposed 8 × 8 2-D DCT core is demonstrated in Section IV. The comparisons and results are presented in Section V, and conclusions are drawn in Section VI.

II. Mathematical Derivation of Distributed Arithmetic

The inner product is an important tool in digital signal processing applications. It can be written as follows:

$$Y = \mathbf{A}^T\mathbf{X} = \sum_{i=1}^{L} A_i X_i \tag{1}$$

where A_i, X_i, and L are ith fixed coefficient, ith input data, and number of inputs, respectively. Assume that coefficient A_i is Q-bit two's complement binary fraction number. Equation (1) can be expressed as follows:

$$\begin{aligned} Y &= \begin{bmatrix} 2^0 & 2^{-1} & \cdots & 2^{-(Q-1)} \end{bmatrix} \\ &\quad \cdot \begin{bmatrix} A_{1,0} & A_{2,0} & \cdots & A_{L,0} \\ A_{1,1} & A_{2,1} & \cdots & A_{L,1} \\ \vdots & \vdots & \ddots & \vdots \\ A_{1,(Q-1)} & A_{2,(Q-1)} & \cdots & A_{L,(Q-1)} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_L \end{bmatrix} \\ &= \begin{bmatrix} 2^0 & 2^{-1} & \cdots & 2^{-(Q-1)} \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{(Q-1)} \end{bmatrix} \end{aligned} \tag{2}$$

where $y_j = \sum_{i=1}^{L} A_{i,j} X_i$, $A_{i,j} \in \{0,1\}$ for $1 \le j \le (Q-1)$, and $A_{i,j} \in \{-1,0\}$ for $j=0$. Note that y_0 may be 0 or a negative number due to two's complement representation. In (2), y_j can be calculated by adding all X_i values when $A_{i,j}=1$, and then the transform output Y can be obtained by shifting and adding all nonzero y_j values. Thus, the inner product computation in (1) can be implemented by using shifting and adders instead of multipliers. Therefore, low hardware cost can be achieved by using DA-based architecture.

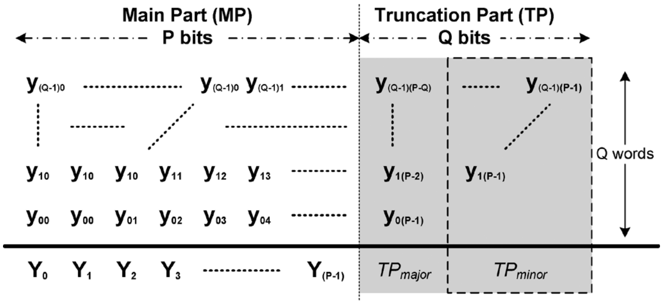


Fig. 1. Q P-bit words shifting and addition operations in parallel.

III. ECAT Architecture

From (2), the shifting and addition computation can be written as follows:

$$Y = \sum_{j=0}^{Q-1} y_j \cdot 2^{-j}. \tag{3}$$

In general, the shifting and addition computation uses a shift-and-add operator [7] in VLSI implementation in order to reduce hardware cost. However, when the number of the shifting and addition words increases, the computation time will also increase. Therefore, the shift-adder-tree (SAT) presented in [10] operates shifting and addition in parallel by unrolling all the words needed to be computed for high-speed applications. However, a large truncation error occurs in SAT, and an ECAT architecture is proposed in this brief to compensate for the truncation error in high-speed applications.

In Fig. 1, the Q P-bit words operate the shifting and addition in parallel by unrolling all computations. Furthermore, the operation in Fig. 1 can be divided into two parts: the main part (MP) that includes P most significant bits (MSBs) and the truncation part (TP) that has Q least significant bits (LSBs). Then, the shifting and addition output can be expressed as follows:

$$Y = \mathrm{MP} + \mathrm{TP} \cdot 2^{-(P-2)}. \tag{4}$$

The output Y will obtain the P-bit MSBs using a rounding operation called post truncation (Post-T), which is used for high-accuracy applications. However, hardware cost increases in the VLSI design. In general, the TP is usually truncated to reduce hardware costs in parallel shifting and addition operations, known as the direct truncation (Direct-T) method. Thus, a large truncation error occurs due to the neglecting of carry propagation from the TP to MP. In order to alleviate the truncation error effect, several error compensation bias methods have been presented [12]–[14]. All previous works were only applied in the design of a fixed-width multiplier. Because the products in a multiplier have a relationship between the input multiplier and multiplicand, the compensation methods usually use the correlation of inputs to calculate a fixed [12] or an adaptive [13], [14] compensation bias using simulation or statistical analysis. Note that the addition elements y_{qp} in the TP in Fig. 1 (where $1 \le q \le (Q-1)$ and $(P-q-1) \le p \le (P-1)$) are independent from each other. Therefore, the previous compensation method cannot be applied in this work, and the proposed ECAT is explained as follows.

A. Proposed Error-Compensated Scheme

From Fig. 1, (4) can be approximated as

$$Y \approx \mathrm{MP} + \sigma \cdot 2^{-(P-2)} \tag{5}$$

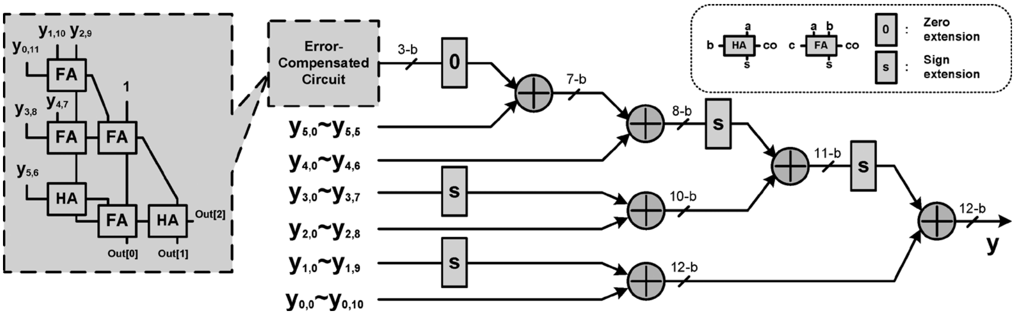


Fig. 2. Proposed ECAT architecture of shifting and addition operators for the $(P,Q)=(12,6)$ example.

where σ is the compensated bias from the TP to the MP as listed in (6)–(8)

$$\sigma=\mathrm{Round}(\mathrm{TP}_{\mathrm{major}}+\mathrm{TP}_{\mathrm{minor}}) \tag{6}$$

and

$$\mathrm{TP}_{\mathrm{major}}=\frac{1}{2}\sum_{j=0}^{Q-1}y_{j(P-1-j)} \tag{7}$$

$$\begin{aligned}\mathrm{TP}_{\mathrm{minor}}=&\frac{1}{4}\left(y_{1(P-1)}+\cdots+y_{(Q-1)(P-Q+1)}\right)\\&+\frac{1}{8}\left(y_{2(P-1)}+\cdots+y_{(Q-1)(P-Q+2)}\right)+\cdots\\&+\left(\frac{1}{2}\right)^{Q}y_{(Q-1)(P-1)}\end{aligned} \tag{8}$$

where $\mathrm{Round}()$ is rounded to the nearest integer. The $\mathrm{TP}_{\mathrm{major}}$ has more weight than $\mathrm{TP}_{\mathrm{minor}}$ when contributing towards the σ. Therefore, the compensated bias σ can be calculated by obtaining $\mathrm{TP}_{\mathrm{major}}$ and estimating $\mathrm{TP}_{\mathrm{minor}}$. Let the probability of $y_{qp}=1$ be 0.5, where $1\le q\le(Q-1)$ and $(P-q-1)\le p\le(P-1)$. Hence, (8) can be expressed as follows:

$$\begin{aligned}\mathrm{TP}_{\mathrm{minor}}&=\frac{1}{4}\left(\frac{1}{2}(Q-1)\right)+\cdots+\left(\frac{1}{2}\right)^{Q+1}\\&=\left(\frac{1}{2}\right)^{Q+2}\sum_{n=1}^{Q-1}n\cdot 2^{n}=\frac{(Q-2)}{4}+\left(\frac{1}{2}\right)^{Q+1}.\end{aligned} \tag{9}$$

For a given $\mathrm{TP}_{\mathrm{major}}$, $(y_{j(P-1-j)}, 0\le j\le(Q-1))$, the σ can be obtained after rounding the sum of $(\mathrm{TP}_{\mathrm{major}}+\mathrm{TP}_{\mathrm{minor}})$. In order to round the summation, $\mathrm{TP}_{\mathrm{minor}}$ can be divided into four parts:

$$\mathrm{TP}_{\mathrm{minor}}=\begin{cases}k-\frac{1}{2}+\left(\frac{1}{2}\right)^{4k+1}, & \text{for } Q=4k\\ k-\frac{1}{4}+\left(\frac{1}{2}\right)^{4k+2}, & \text{for } Q=4k+1\\ k+\left(\frac{1}{2}\right)^{4k+3}, & \text{for } Q=4k+2\\ k+\frac{1}{4}+\left(\frac{1}{2}\right)^{4k+4}, & \text{for } Q=4k+3.\end{cases} \tag{10}$$

As $k\ge 1$, the $\mathrm{TP}_{\mathrm{minor}}$ approximates (11)

$$\mathrm{TP}_{\mathrm{minor}}\approx\begin{cases}(k-1)+\frac{1}{2}, & \text{for } Q=4k\\ (k-1)+\frac{3}{4}, & \text{for } Q=4k+1\\ k, & \text{for } Q=4k+2\\ k+\frac{1}{4}, & \text{for } Q=4k+3.\end{cases} \tag{11}$$

Hence, σ can be rewritten as three cases.

Case 1) $Q=0,1,2,3$

$$\sigma=\mathrm{Round}(\mathrm{TP}_{\mathrm{major}}). \tag{12}$$

Case 2) $Q=4k, 4k+1\ (k\ge 1)$

$$\sigma=(k-1)+\mathrm{Round}(\mathrm{TP}_{\mathrm{major}}+0.5). \tag{13}$$

TABLE I
COMPARISONS OF ABSOLUTE AVERAGE ERROR ε, MAXIMUM ABSOLUTE ERROR $\varepsilon_{\max}$, AND MEAN SQUARE ERROR $\varepsilon_{\mathrm{mse}}$

Error	$(\mathbf{P},\mathbf{Q})$	$(\mathbf{12},\mathbf{3})$ *case1*	$(\mathbf{12},\mathbf{6})$ *case3*	$(\mathbf{12},\mathbf{9})$ *case2*	$(\mathbf{12},\mathbf{12})$ *case2*
ε	Direct-T	1.0625	2.5078	4.0010	5.5001
	Proposed	***0.2656***	***0.3789***	***0.3804***	***0.4738***
	Post-T	0.2500	0.2500	0.2500	0.2500
$\varepsilon_{\max}$	Direct-T	2.1250	5.0156	8.0020	11.000
	Proposed	***0.6250***	***1.5000***	***2.0020***	***3.0000***
	Post-T	0.5000	0.5000	0.5000	0.5000
ε_{mse}	Direct-T	1.3516	6.7614	16.730	31.224
	Proposed	***0.1016***	***0.2184***	***0.2222***	***0.3472***
	Post-T	0.0859	0.0834	0.0833	0.0833

Case 3) $Q=4k+2, 4k+3\ (k\ge 1)$

$$\sigma=k+\mathrm{Round}(\mathrm{TP}_{\mathrm{major}}). \tag{14}$$

B. Performance Simulation for an Error-Compensated Circuit

In this subsection, comparisons of the absolute average error ε, the maximum error $\varepsilon_{\max}$, and the mean square error $\varepsilon_{\mathrm{mse}}$ for the proposed error-compensated circuit with Direct-T and Post-T are listed in Table I. The ε, $\varepsilon_{\max}$, and $\varepsilon_{\mathrm{mse}}$ are defined as follows:

$$\varepsilon=Avg\left\{|\mathrm{TP}-\sigma|\right\} \tag{15}$$

$$\varepsilon_{\max}=\max\left\{|\mathrm{TP}-\sigma|\right\} \tag{16}$$

$$\varepsilon_{mse}=Avg\left\{(\mathrm{TP}-\sigma)^{2}\right\} \tag{17}$$

where $Avg\{\}$ is the average operator.

The internal word-length usually uses 12 bits in a DCT design. Consequently, word length $P=12$ is chosen together with different Q values of 3, 6, 9, and 12, which are listed in Table I. The Post-T method provides the most accurate values for fixed-width computation nowadays. In addition, the Direct-T method has the largest inaccuracies of the errors shown in Table I for low-cost hardware design. The proposed ECAT is more accurate than Direct-T and is close to the performance of the Post-T method using a compensated circuit. Because the truncation part $\mathrm{TP}_{\mathrm{minor}}$ is estimated using statistical analysis, the magnitude of errors also increases as the number of shift-and-add words Q increases.

C. Proposed ECAT Architecture

The proposed ECAT architecture is illustrated in Fig. 2 for $(P,Q)=(12,6)$ (*case 3*), where block FA indicates a full-adder cell with three inputs (a, b, and c) and two outputs, a sum (s) and a carry-out (co). Also,

TABLE II
COMPARISONS OF THE PROPOSED ECAT WITH OTHER ARCHITECTURES FOR A SIX 8-BIT WORDS EXAMPLE

	Shift-and-add	SAT	Proposed ECAT
Area (gates)	236	406	463
Delay (ns)	10.8	3.72	3.89
Area×delay	100 %	59.3 %	70.7 %
ε_{mse}	0.326	6.761	0.218

TABLE III
9-BIT DA-BASED COEFFICIENT MATRIX $\mathbf{C}_{ee}$

Z_0		Z_4	
weight	value	weight	value
-2^0	0	-2^0	A_1
2^{-1}	$A_0 + A_1$	2^{-1}	A_0
2^{-2}	0	2^{-2}	A_1
2^{-3}	$A_0 + A_1$	2^{-3}	A_0
2^{-4}	$A_0 + A_1$	2^{-4}	A_0
2^{-5}	0	2^{-5}	A_1
2^{-6}	$A_0 + A_1$	2^{-6}	A_0
2^{-7}	0	2^{-7}	A_1
2^{-8}	$A_0 + A_1$	2^{-8}	$A_0 + A_1$

block HA indicates half-adder cell with two inputs (a and b) and two outputs, a sum (s) and a carry-out (co). The comparisons of area, delay, area-delay product, and accuracy for the proposed ECAT with other architectures are listed in Table II. The area and delay are synthesized using a Synopsys Design Compiler with the Artisan TSMC 0.18-μm Standard cell library.

The proposed ECAT has the highest accuracy with a moderate area-delay product. The shift-and-add [7] method has the smallest area, but the overall computation time is equal to $10.8(= 1.8 \times 6)$ ns that is the longest. Similarly, the SAT [10], which truncates the TP and computes in parallel, takes 3.72 ns to complete the computation and uses 406 gates, which is the best area-delay product performance. However, for system accuracy, the SAT is the worst option shown in Table II. Therefore, the ECAT is suitable for high-speed and low-error applications.

IV. PROPOSED 8 × 8 2-D DCT CORE DESIGN

The 1-D DCT employs the DA-based architecture and the proposed ECAT to achieve a high-speed, small area, and low-error design. The 1-D 8-point DCT can be expressed as follows:

$$Z_n = \frac{1}{2} k_n \sum_{m=0}^{7} x_m \times \cos\left(\frac{(2m+1)n\pi}{16}\right) \qquad (18)$$

where x_m denotes the input data; Z_n denotes the transform output; $0 \leq n \leq 7$; $k_n = 1/\sqrt{2}$ for $n = 0$; and $k_n = 1$ for other n values. By neglecting the scaling factor 1/2, the 1-D 8-point DCT in (18) can be divided into even and odd parts: $\mathbf{Z}_e$ and $\mathbf{Z}_o$ as listed in (19) and (20), respectively

$$\mathbf{Z}_e = \begin{bmatrix} Z_0 \\ Z_2 \\ Z_4 \\ Z_6 \end{bmatrix} = \begin{bmatrix} c_4 & c_4 & c_4 & c_4 \\ c_2 & c_6 & -c_6 & -c_2 \\ c_4 & -c_4 & -c_4 & c_4 \\ c_6 & -c_2 & c_2 & -c_6 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \mathbf{C}_e \cdot \mathbf{a} \qquad (19)$$

$$\mathbf{Z}_o = \begin{bmatrix} Z_1 \\ Z_3 \\ Z_5 \\ Z_7 \end{bmatrix} = \begin{bmatrix} c_1 & c_3 & c_5 & c_7 \\ c_3 & -c_7 & -c_1 & -c_5 \\ c_5 & -c_1 & c_7 & c_3 \\ c_7 & -c_5 & c_3 & -c_1 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \mathbf{C}_o \cdot \mathbf{b} \qquad (20)$$

where $c_i = \cos(i\pi/16)$. Moreover, the even part $\mathbf{Z}_e$ can be further decomposed into even and odd parts: $\mathbf{Z}_{ee}$ and $\mathbf{Z}_{eo}$

$$\mathbf{Z}_{ee} = \begin{bmatrix} Z_0 \\ Z_4 \end{bmatrix} = \begin{bmatrix} c_4 & c_4 \\ c_4 & -c_4 \end{bmatrix} \begin{bmatrix} A_0 \\ A_1 \end{bmatrix} = \mathbf{C}_{ee} \cdot \mathbf{A} \qquad (21)$$

$$\mathbf{Z}_{eo} = \begin{bmatrix} Z_2 \\ Z_6 \end{bmatrix} = \begin{bmatrix} c_2 & c_6 \\ c_6 & -c_2 \end{bmatrix} \begin{bmatrix} B_0 \\ B_1 \end{bmatrix} = \mathbf{C}_{eo} \cdot \mathbf{B}. \qquad (22)$$

For the DA-based computation, the coefficient matrix $\mathbf{C}_o$, $\mathbf{C}_{ee}$, and $\mathbf{C}_{eo}$, are expressed as 9-bit binary fraction numbers. Table III expresses $\mathbf{Z}_{ee}$ (Z_0 and Z_4) in the bit level formulation. In Table III, using given input data A_0 and A_1, the transform output $\mathbf{Z}_{ee}$ needs only one adder to compute $(A_0 + A_1)$ and two separated ECATs to obtain the results of Z_0 and Z_4. Similarly, the other transform outputs $\mathbf{Z}_{eo}$ and $\mathbf{Z}_o$ can be implemented in DA-based forms using $10(= 1 + 9)$ adders and corresponding ECATs. Consequently, from the (19)–(22), the proposed 1-D 8-point DCT architecture can be constructed as illustrated in Fig. 3 using a DA-Butterfly-Matrix, that includes two DA even processing elements (DAEs), a DA odd processing element (DAO) and 12 adders/subtractors, and 8 ECATs (one ECAT for each transform output Z_n). The eight separated ECATs work simultaneously, enabling high-speed applications to be achieved. After the data output from the DA-Butterfly-Matrix is completed, the transform output $\mathbf{Z}$ will be completed during one clock cycle by the proposed ECATs. In contrast, the traditional shift-and-add architecture requires Q clock cycles to complete the transform output $\mathbf{Z}$ if the DA-precision is Q bits.

With high-speed considerations in mind, the proposed 2-D DCT is designed using two 1-D DCT cores and one transpose buffer. For accuracy, the DA-precision and transpose buffer word lengths are chosen to be 9 bits and 12 bits, respectively, meaning that the system can meet the PSNR requirements outlined in previous works. Moreover, the 2-D DCT core accepts 9-bit image input and 12-bit output precision.

For the proposed 2-D DCT, the Synopsys Design Compiler was applied to synthesize the RTL design of the proposed core, and the Cadence SoC Encounter was adopted for placement and routing (P&R). Implemented in a 1.8-V TSMC 0.18-μm 1P6M CMOS process, the proposed 8 × 8 2-D DCT core has a latency of 10 clock cycles and is operated at 125 MHz. As a result of the 8 parallel outputs, the proposed 2-D DCT core can achieve a throughput rate of 1 Gpixels per second (= 8× 125 MHz), meeting the 1080 p (1920×1080 × 60 pixels/s) high definition television (HDTV) specifications for 200 MHz based on low power operations. The core layout and simulated characteristics are shown in Fig. 4.

V. DISCUSSION AND COMPARISONS

The test image "Lena" used to check system accuracy is comprised of 512 × 512 pixels with each pixel being represented by 8-bit 256 gray level data. After inputting the original test image pixels to the proposed 2-D DCT core, the transform output data is captured and fed into MATLAB to compute the inverse DCT using 64-bit double-precision operations. The PSNRs are close to 44 and 47 dB for test image and for random 8-bit 256 gray level data inputs, receptively.

Table IV compares the proposed 8 × 8 2-D DCT core with previous 2-D DCT cores. In [3], a multiplier-based DCT core based on pipeline radix-4^2 single delay feedback path ($R4^2\mathrm{SDF}$) architecture to achieve high-speed design. The ROM-based DCT core is presented in [4] to reduce hardware cost. In [7], a NEDA architecture is presented by using

TABLE IV
COMPARISONS OF DIFFERENT 2-D DCT ARCHITECTURES WITH THE PROPOSED ARCHITECTURE

	Lin et al. [3]	Uramoto et al. [4]	Shams et al. [7]	Chungan et al. [10]	Huang et al. [11]	**Proposed**
Architecture	Multiplier-based	ROM-based	NEDA	DA-based	DA-based	**DA-based**
Technology	0.13μmm	0.8μm	0.18μm	0.18μm	0.18μm	**0.18μm**
Multipliers/ROMs	1/0	0/256	0/0	0/0	0/0	**0/0**
Adders	26	16	92	12+18ALU*+16SAT	50+16AT	**46+16 ECAT***
DA-precision	–	–	12 bits	13 bits	13 bits	**9 bits**
Throughput Rate (pels/sec)	100 M	100 M	77 M†	250 M	400 M	**1 G**
Gate Counts(NAND2)‡	60 K	25.5 K	22.5 K	27.8 K	39.8 K	**22.2 K**
Hardware Efficiency	1.6	3.92	3.42	9	10.05	**45**
Accuracy (CCITT$^\diamond$ Compatible)	Yes	Yes	Yes	N/A	N/A	**Yes**

* ALU: Arithmetic logic unit. ‡ 4 transistors per NAND2 gate for different technology. $^\diamond$ CCITT: Consultative Committee for International Telegraph and Telephone.

* ECAT: The proposed error-compensated adder-tree. † 77 MHz $= 1$ GHz$/13$, where denominator 13 is the number of shifting and addition computation cycles.

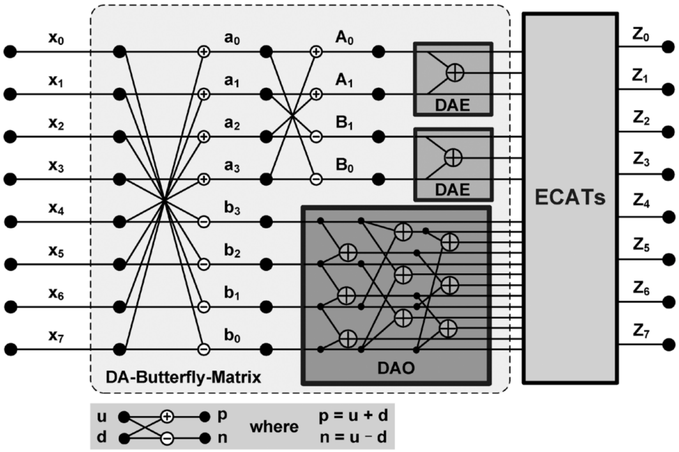
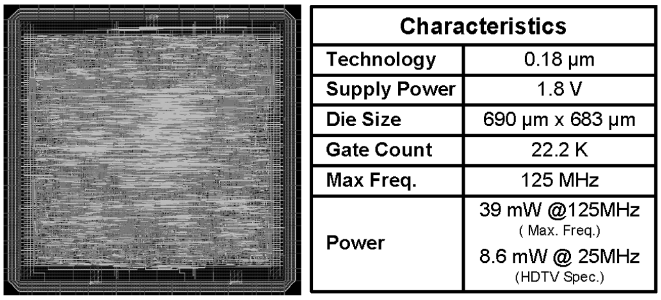


Fig. 3. Architecture of the proposed 1-D 8-point DCT.



Characteristics	
Technology	0.18 µm
Supply Power	1.8 V
Die Size	690 µm x 683 µm
Gate Count	22.2 K
Max Freq.	125 MHz
Power	39 mW @125MHz (Max. Freq.) 8.6 mW @ 25MHz (HDTV Spec.)

Fig. 4. Core layout and characteristics.

adders to reduce the chip area of DCT core. Nevertheless, a speed limitation for shift-and-add is in NEDA design. In [10] and [11], the SAT and AT architectures for DA-based DCTs improve the throughput rate of the NEDA method. However, DA-precision must be chosen as 13 bits to meet the system accuracy with more area overhead. The proposed DCT core uses low-error ECAT to achieve a high-speed design, and the DA-precision can be chosen as 9 bits to meet the PSNR requirements for reducing hardware costs. The proposed DCT core has the highest hardware efficiency, defined as follows (based on the accuracy required by the presented standards)

$$\text{Hardware Efficiency}(10^3 \text{ pels/s}) = \frac{\mathrm{ThroughputRate}}{\mathrm{Gate\ Counts}}. \tag{23}$$

TABLE V
COMPARISONS OF 2-D DCT ARCHITECTURES IN FPGAS

FPGA-Chip	XC2VP30			XC3S200	
Architecture	[15]	[16]	Proposed	[17]	Proposed
# of 4 input LUTs	10310	2618	2990	2271	2847
# of Slices	5729	2823	1872	1221	1585
# of Slice Flip Flops	3736	3431	1837	616	1817
Clock Freq. (MHz)	149	107	99	50	61
Throughput (M-pels/s)	149	107	792	400	488
Power (mW)	N/A	N/A	166.8	281	91

Furthermore, the proposed 2-D DCT core synthesized by using Xilinx ISE 9.1, and the Xilinx XC2VP30 FPGA can achieve 792 mega pixels per second (M-pels/sec) throughput rate (up to about 7 folds of previous work [16]). Table V compares the proposed 2-D DCT core with previous FPGA implementations.

VI. CONCLUSION

In this brief, a high-speed and low-error 8 × 8 2-D DCT design with ECAT is proposed to improve the throughput rate significantly up to about 13 folds at high compression rates by operating the shifting and addition in parallel. Furthermore, the proposed error-compensated circuit alleviates the truncation error in ECAT. In this way, the DA-precision can be chosen as 9 bits instead of 12 bits so as to meet the PSNR requirements. Thus, the proposed DCT core has the highest hardware efficiency than those in previous works for the same PSNR requirements. Finally, an area-efficient 2-D DCT core is implemented using a TSMC 0.18-μm process, and the maximum throughput rate is 1 Gpels/s. In summary, the proposed architecture is suitable for high compression rate applications in VLSI designs.

REFERENCES

[1] Y. Wang, J. Ostermann, and Y. Zhang, *Video Processing and Communications*, 1st ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.

[2] Y. Chang and C. Wang, "New systolic array implementation of the 2-D discrete cosine transform and its inverse," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 2, pp. 150–157, Apr. 1995.

[3] C. T. Lin, Y. C. Yu, and L. D. Van, "Cost-effective triple-mode reconfigurable pipeline FFT/IFFT/2-D DCT processor," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 16, no. 8, pp. 1058–1071, Aug. 2008.

[4] S. Uramoto, Y. Inoue, A. Takabatake, J. Takeda, Y. Yamashita, H. Yerane, and M. Yoshimoto, "A 100-MHz 2-D discrete cosine transform core processor," *IEEE J. Solid-State Circuits*, vol. 27, no. 4, pp. 492–499, Apr. 1992.

[5] S. Yu and E. E. S. , Jr., "DCT implementation with distributed arithmetic," *IEEE Trans. Comput.*, vol. 50, no. 9, pp. 985–991, Sep. 2001.

[6] P. K. Meher, "Unified systolic-like architecture for DCT and DST using distributed arithmetic," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, no. 12, pp. 2656–2663, Dec. 2006.

[7] A. M. Shams, A. Chidanandan, W. Pan, and M. A. Bayoumi, "NEDA: A low-power high-performance DCT architecture," *IEEE Trans. Signal Process.*, vol. 54, no. 3, pp. 955–964, Mar. 2006.

[8] M. R. M. Rizk and M. Ammar, "Low power small area high performance 2D-DCT architecture," in *Proc. Int. Design Test Workshop*, 2007, pp. 120–125.

[9] Y. Chen, X. Cao, Q. Xie, and C. Peng, "An area efficient high performance DCT distributed architecture for video compression," in *Proc. Int. Conf. Adv. Comm. Technol.*, 2007, pp. 238–241.

[10] C. Peng, X. Cao, D. Yu, and X. Zhang, "A 250 MHz optimized distributed architecture of 2D 8 × 8 DCT," in *Proc. Int. Conf. ASIC*, 2007, pp. 189–192.

[11] C. Y. Huang, L. F. Chen, and Y. K. Lai, "A high-speed 2-D transform architecture with unique kernel for multi-standard video applications," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2008, pp. 21–24.

[12] S. S. Kidambi, F. E. Guibaly, and A. Antonious, "Area-efficient multipliers for digital signal processing applications," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 43, no. 2, pp. 90–95, Feb. 1996.

[13] K. J. Cho, K. C. Lee, J. G. Chung, and K. K. Parhi, "Design of low-error fixed-width modified booth multiplier," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 12, no. 5, pp. 522–531, May 2004.

[14] L. D. Van and C. C. Yang, "Generalized low-error area-efficient fixed-width multipliers," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 52, no. 8, pp. 1608–1619, Aug. 2005.

[15] C. C. Sun, P. Donner, and J. Gotze, "Low-complexity multi-purpose IP core for quantized discrete cosine and integer transform," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2009, pp. 3014–3017.

[16] A. Tumeo, M. Monchiero, G. Palermo, F. Ferrandi, and D. Sciuto, "A pipelined fast 2D-DCT accelerator for FPGA-based SoCs," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI*, 2007, pp. 331–336.

[17] S. Ghosh, S. Venigalla, and M. Bayoumi, "Design and implementaion of a 2D-DCT architecture using coefficient distributed arithmetic," in *Proc. IEEE Comput. Soc. Ann. Symp. VLSI*, 2005, pp. 162–166.

Landauer Clocking for Magnetic Cellular Automata (MCA) Arrays

Anita Kumari and Sanjukta Bhanja

***Abstract*—Magnetic cellular automata (MCA) is a variant of quantum-dot-cellular automata (QCA) where neighboring single-domain nanomagnets (also termed as magnetic cell) process and propagate information (logic 1 or logic 0) through mutual interaction. The attractive nature of this framework is that not only room temperature operations are feasible but also interaction between neighbors is central to information processing as opposed to creating interference. In this work, we explore spatially moving Landauer clocking scheme for MCA arrays (length of 8, 16, and 32 cells) and show the role and effectiveness of the clock in propagating logic signal from input to output without magnetic frustration. Simulation performed in object oriented micromagnetic framework suggests that the clocking field is sensitive to scaling, shape, and aspect ratio.**

***Index Terms*—Clock, magnetic cellular automata (MCA), quantum-dot-cellular automata (QCA).**

Manuscript received February 13, 2009; revised July 16, 2009. First published January 22, 2010; current version published March 23, 2011. This work was supported in part by National Science Foundation Career Award 0639624, by the National Science Foundation Computing Research Infrastructure (CRI) Grant 0551621, and by the National Science Foundation Emerging Models for Technology (EMT) Grant 0829838.

The authors are with the Nano Computing Research Group (NCRG), Department of Electrical Engineering, University of South Florida, Tampa, FL 33620 USA (e-mail: akumari@mail.usf.edu; bhanja@eng.usf.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2009.2036627

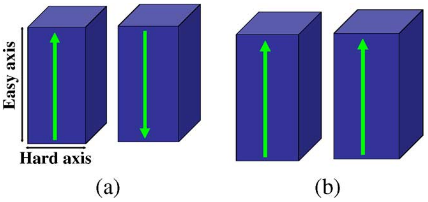


Fig. 1. (a) Definition of logic "1" and "0" for nanomagnets. (b) Metastable states for coupled pairs.

I. Introduction

One of the pioneering efforts in field-coupled cellular automata computing evolved using quantum tunneling interactions of electrons in neighboring cell [1] is the promising phenomenal packing density, and the low power-delay product. In this work, we study magnetic cellular automata architecture, already functionally demonstrated by pioneering efforts of Imre *et al.* [2], [3] that promise stable operation at room temperature alleviating the criticism of some of the other Cellular Automata variations. The salient feature of the magnetic cellular automata architectures are: 1) single-domain structure and the shape anisotropy work magnificently to store Boolean logic as the easy axis (Y-axis in our case) magnetization; 2) magnetic coupling between the interacting neighbors assures anti-ferromagnetic alignments (anti-parallel), thus generating the signal and its inverse next to each other [see Fig. 1(a)] and; 3) since magnetic interactions are direction-insensitive, we need an addition control apart from input to drive the information flow from input to output which is commonly termed as *clock*. We have observed that conventional adiabatic clock, having group of nanomagnets in one clock state [4] does not work well for lengthy magnetic cellular automata (MCA) array. So we propose a spatially moving clock field named as Landauer clock, accomplished by magnetically switching cell from a null state [the state which holds no binary information ("1" or "0")], through a switching state (in which the nanomagnet state is determined by its neighbor) and finally to a locked state (stable state) (in which the state is independent of the previous neighbor).

We used a micro-magnetic simulator [object oriented micromagnetic framework (OOMMF)] that solves the Landau–Lifshitz equations accounting various energies (zeeman energy, magnetostatic energy, exchange energy, anisotropy energy, demagnetization energy, etc.). We demonstrated the spatial temporal clock known as Landauer clock on different length arrays (8, 16, 32), different shapes (rectangular and oval) and different nanomagnet aspect ratio (AR). The aspect ratio is the width to height ratio.

A few observations made by our experimental simulations for the clocking scheme are as follows.

1) Clock field is invariant with length (8, 16, and 32) and works perfectly all the time, yielding anti-parallel cell.
2) Oval shape nanomagnet requires high clock field strength due to high coercivity as compared to rectangular shape nanomagnet. Hence it is not suitable for MCA architecture.
3) Input field required is very low as compared to the null and switch fields and is same for both shapes (rectangle and oval) for aspect ratios under study.
4) Clock field decreases linearly with scaling of nanomagnet.

II. Theoretical Background

Magnetic field coupling is emerging as a promising successor of CMOS. The behavior of magnetic materials is described by the classical theory of micromagnetism. In bulk materials the balance of

1063-8210/$26.00 © 2010 IEEE