Generalized Low-Error Area-Efficient Fixed-Width Multipliers

Lan-Da Van, *Member, IEEE,* and Chih-Chyau Yang

***Abstract*—In this paper, we extend our previous methodology for designing a family of low-error area-efficient fixed-width two's-complement multipliers that receive two n-bit numbers and produce an n-bit product. The generalized methodology involving four steps results in several better error-compensation biases. These better error-compensation biases can be easily mapped to low-error area-efficient fixed-width multipliers suitable for very large-scale integration implementation and digital signal processing application. Via the proposed Type 1 8×8 fixed-width multiplier, the reduction of the average error can be improved by 88% compared with the direct-truncated (D-Truncated) multiplier. It is also shown that the same proposed multiplier leads to 32.75% reduction in area compared with the standard multiplier.**

***Index Terms*—Area efficient, Baugh–Wooley algorithm, fixed-width multiplier, truncation error.**

I. Introduction

DURING THE PAST decade, digital signal processing (DSP) kernels stand the test for widely multimedia-communication applications. It is desirable to operate fixed-width (also referred to as single precision) multiplication with low truncation error [1]–[8] among many DSP kernels such as digital filters [3], [8]–[10] and wavelet transform [7]. The fixed-width multipliers derived from Baugh–Wooley [11]–[13] multiplier produce n-bit output product with n-bit multiplier and n-bit multiplicand. Area saving of a fixed-width multiplier can be achieved either by directly truncating n least significant columns and preserving n most significant columns or by other efficient methods [1]–[8]. By the former method, significant truncation errors will be introduced since no error compensation is considered. Thus, the latter schemes explore issues on low truncation error and small area. Lim [1] first utilized statistical techniques to estimate and simulate the error-compensation bias. However, in his analysis, the reduction and rounding errors are separately treated such that this scheme does not lead to an accurate enough error-compensation bias. Reduction error occurs because the $n - w$ least signification columns of the subproduct array are not used to compute the product, where w is a nonnegative integer between 0 and $n - 1$. Rounding error occurs because the product is rounded to n bits. Note that the sum of the reduction and rounding errors equals the truncation error [2]. Schulte *et al.* [2] improved the error-compensation bias to be more accurate and practical since the reduction and rounding errors are concurrently treated. The above two schemes are based on keeping $n + w$ most significant columns of the subproduct array. While w equals zero, these two schemes are equivalent to the work of Kidambi *et al.* [3]. Nevertheless, the three analyzes and structures cannot provide an adaptive error-compensation bias by taking account of unused product-bits in hardware. Later, King and Swartzlander [4]–[6] analyzed an adaptive error-compensation bias (also referred to as variable correction) under keeping $n + w$ most significant columns and proposed an n-bit fixed-width multiplier. Corresponding to J-K-Cs' index, Jou *et al.* [7], independently, provided another adaptive error-compensation bias (also referred to as carry-generating circuit) to improve truncation error at $w = 0$. In [8], how to choose the index θ_{index} and one specific low-error fixed-width multiplier structure for $w = 0$ have been partially investigated and pointed out, respectively. This work is intended to extend our previous methodology to design several low-error area-efficient fixed-width two's-complement multipliers under different values of w. Note that the features of low error and area efficiency of our proposed multipliers can be revealed through the comparison with the direct-truncated (D-Truncated) multiplier and standard multiplier, respectively. This generalized methodology includes the following phases in order:

1) propose an error-compensation bias with a new binary thresholding for a fixed value of w;
2) simulate the value of K and error performance of the proposed error-compensation bias using our generalized index, and then select the best index having lower error and satisfying the same value of K for limited width n;
3) verify the realizable error compensation bias by statistical techniques;
4) construct the low-error fixed-width multiplier structure.

In this paper, the limited width is under the range of $n \leq 16$. The realization of this methodology and the detailed terminologies will be debated in Section III. Through the verification of the chip implementation, these low-error multipliers at $w = 0$ are shown to have the same chip-area ratio as J-K-Cs' multiplier. While $w \geq 1$, the new multiplier also operates lower error than those in [4]–[6] at the expense of slightly increased area-ratio with respect to each value of w. Thus, the proposed low-error fixed-width multipliers are area-efficient.

This paper is organized as follows. The Baugh–Wooley array multiplier is briefly reviewed in Section II. In Section III, by properly choosing binary thresholding as well as the generalized index, we propose several better error-compensation biases and simulate the results for $n \leq 16$. And then, we show that the new

Manuscript received December 28, 2003; revised June 3, 2004 and October 13, 2004. This paper was presented in part at the 2002 International Symposium on Circuits and Systems (ISCAS), AZ, USA. This work was supported by the Chip Implementation Center (CIC) and by National Applied Research Laboratories. This paper was recommended by Associate Editor Z. Wang.

The authors are with the Chip Implementation Center (CIC), National Applied Research Laboratories, Hsinchu 300, Taiwan, R.O.C. (e-mail: ldvan@cic.org.tw).

Digital Object Identifier 10.1109/TCSI.2005.851675

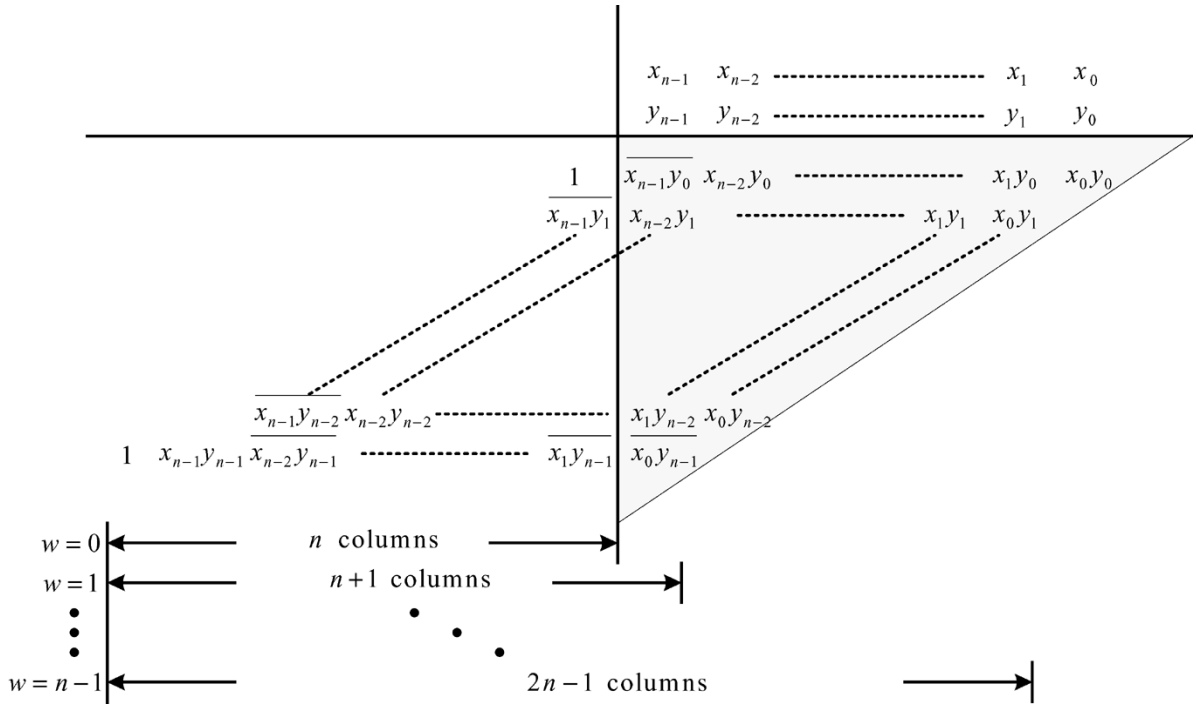

Fig. 1. Subproduct array of $n \times n$ two's-complement multiplication.

error-compensation biases still hold for each value of n by using statistical techniques. The improved error-compensation bias can be mapped to a new structure with respect to each value of w. The comparison results in terms of relative maximum error, relative average error, relative variance of the error and chip-area ratio in percentage are presented in Section IV. For speech processing application, the error performance using the various fixed-width multipliers is clearly shown in the same section. At last, brief statements conclude the presentation of this paper.

II. Fundamentals of Baugh–Wooley Array Multiplier

Considering two two's-complement integer operands, an n-bit multiplicand X and an n-bit multiplier Y can be respectively represented by

$$X = -x_{n-1}2^{n-1} + \sum_{i=0}^{n-2} x_i 2^i \tag{1}$$

$$Y = -y_{n-1}2^{n-1} + \sum_{j=0}^{n-2} y_j 2^j \tag{2}$$

where $x_i,\ y_j \in \{0,1\}$. The $2n$-bit product P_{Standard} can be written as

$$\begin{aligned} P_{\text{Standard}} &= X \times Y \\ &= x_{n-1}y_{n-1}2^{2n-2} + \sum_{i=0}^{n-2}\sum_{j=0}^{n-2} x_i y_j 2^{i+j} \\ &\quad + 2^{n-1}\left(-2^{n-1} + \sum_{j=0}^{n-2} \overline{x_{n-1}y_j}2^j + 1\right) \\ &\quad + 2^{n-1}\left(-2^{n-1} + \sum_{i=0}^{n-2} \overline{y_{n-1}x_i}2^i + 1\right). \end{aligned} \tag{3}$$

Equation (3) represents the Baugh–Wooley algorithm [11]–[13] in which this array multiplier sums partial-product bits corresponding to each weighting. Fig. 1 reveals the subproduct array for $n \times n$ two's-complement multiplication.

III. Low-Error Fixed-Width Multipliers

Since partial products for Baugh–Wooley array multiplier can be partitioned into two sections, (3) can be rewritten as

$$P_{\text{Standard}} = MP + LP = \sum_{i=0}^{2n-1} P_i 2^i \tag{4}$$

where $P_i \in \{0,1\}$, MP is the most significant columns, and LP is the least significant columns as shown in the shadow region of Fig. 1. It is known that the most accurate truncated product is theoretically given by

$$P_{\text{Standard}} \cong MP + \sigma_{\text{Temp}} \times 2^n \tag{5}$$

$$\begin{aligned} \sigma_{\text{Temp}} &= \left[\frac{LP}{2^n}\right]_r \\ &= \left[\frac{1}{2}\left(\overline{x_{n-1}y_0} + x_{n-2}y_1 + \ldots + x_1y_{n-2} + \overline{x_0y_{n-1}}\right)\right. \\ &\quad + \frac{1}{2^2}(x_{n-2}y_0 + \ldots + x_0y_{n-2}) + \ldots \\ &\quad \left. + \frac{1}{2^{n-1}}(x_1y_0 + x_0y_1) + \frac{1}{2^n}x_0y_0\right]_r \end{aligned} \tag{6}$$

where $[\bullet]_r$ denotes the rounding integer of $\bullet$. From (6), it is observed that σ_{Temp} is mainly affected by $\overline{x_{n-1}y_0} + x_{n-2}y_1 + \ldots + x_1y_{n-2} + \overline{x_0y_{n-1}}$ due to the largest weight. Thus, it is convenient to define the main-error compensation term E_{main} and remain-error compensation term E_{remain}. Although E_{main}

and E_{remain} have been defined in [8], for convenience of presentation, we reproduce two terms, respectively, as

$$E_{\text{main}} = \overline{x_{n-1}y_0} + x_{n-2}y_1 + x_{n-3}y_2 + \cdots + x_1y_{n-2} + \overline{x_0y_{n-1}} \quad (7)$$

$$E_{\text{remain}} = \frac{1}{2}(x_{n-2}y_0 + x_{n-3}y_1 + \ldots + x_0y_{n-2}) + \ldots + \frac{1}{2^{n-1}}x_0y_0. \quad (8)$$

Using (7) and (8), we can recast (6) as

$$\sigma_{\text{Temp}} = \left[\frac{1}{2}(E_{\text{main}} + E_{\text{remain}})\right]_r. \quad (9)$$

It should be emphasized that σ_{Temp} is the most accurate error-compensation bias (also called true rounding approach) and the fixed-width two's-complement multiplier structure can be modified from [8, Fig.2 (a)]. The modification requires that n half adders are added in the bottom row of [8, Fig.2 (a)] and then P_7 is directly connected to the input of a half adder in the lower right-hand corner. However, it is apparent that this true rounding approach results in larger area (i.e., higher cost) than that of a standard multiplier. On the contrary, directly truncating LP columns leads to the smaller area fixed-width two's-complement multiplier, but this approach incurs the largest truncation error.

Before extending our proposed systematic methodology, some useful terminologies are indicated here. To explore the influence of the index in the proposed binary thresholding for different values of w, we first define a generalized index, $\theta_{\text{index},w}$, as

$$\theta_{\text{index},w}(q_{n-1-w}, q_{n-2-w}, \ldots, q_0) = \langle x_{n-1-w}y_0\rangle^{q_{n-1-w}} + \langle x_{n-2-w}y_1\rangle^{q_{n-2-w}} + \ldots + \langle x_0y_{n-1-w}\rangle^{q_0} \quad (10)$$

where w means to keep $n+w$ most significant columns of the subproduct array as shown in Fig. 1 and the binary parameters $q_{n-1-w}, q_{n-2-w}, \ldots$, and $q_0 \in \{0,1\}$. The operator

$$\langle T\rangle^{q_i} = \begin{cases} T, & \text{if } q_i = 0 \\ \overline{T}, & \text{if } q_i = 1 \end{cases} \quad (11)$$

in which $\overline{T}$ is the complement of the binary number T. Instead, the index $\theta_{\text{index},w}(q_{n-1-w}, q_{n-2-w}, \ldots, q_0)$ is referred to as $\theta_{Q,w}$, where

$$Q = q_{n-1-w} \times 2^{n-1-w} + q_{n-2-w} \times 2^{n-2-w} + \ldots + q_0 \times 2^0. \quad (12)$$

Note that Q has a range varying from 0 to $2^{n-w}-1$; for example, $\theta_{\text{index},w=0}(10\,000\,001)$ denoted as $\theta_{Q=129,w=0}$ means the 129th index under keeping n columns for $n=8$. The intuitive meaning of $\theta_{Q,w}$ is to properly partition the errors into two groups and the resulting error can be alleviated. In other words, if $\theta_{Q,w}$ cannot be properly chosen, the resulting error will become larger. How to apply $\theta_{Q,w}$ will be demonstrated in this section latter. For evaluating the resulting performance, let ε, $\varepsilon_{\max}$, $\overline{\varepsilon}$ and υ be the absolute error between products of the standard multiplier and truncated multiplier, the maximum error, the average error, and the variance of error, respectively. Since these error terminologies are well defined in [8], herein, we do not repeat the definitions for conciseness. In the following, we divide the content into two subsections.

A. Realizable Error-Compensation Bias Keeping n Most Significant Columns ($w=0$)

In this situation, since w is equal to zero, the most significant columns of reduction error and rounding error occur in the same column; thus, the analyzes of Lim [1] and Schulte *et al.* [2] could be reduced to that of Kidambi *et al.* [3]. The contribution of [3] is to provide a constant error-compensation bias derived from the carry-propagation probability of LP. The truncated multiplier in [3] yields n-bit product P_{K-G-A} as

$$P_{\text{Standard}} \cong P_{K-G-A} = MP + \sigma_{K-G-A} \times 2^n \quad (13)$$

where σ_{K-G-A} represents the error-compensation bias and only depends on the width n. While the width n is given, one can obtain a constant error-compensation bias under uniform distribution input. Note that the analysis of King and Swartzlander [4], [5] does not belong to this subsection since they focus on the design for $w \geq 1$. Then, Jou *et al.* [7] improve truncation error in [3] and suggest a specific truncated multiplier that results in the fixed-width product P_{J-K-C} as

$$P_{\text{Standard}} \cong P_{J-K-C} = MP + \sigma_{J-K-C} \times 2^n \quad (14)$$

$$\sigma_{J-K-C} = \begin{cases} x_{n-2}y_1 + x_{n-3}y_2 + \cdots + x_1y_{n-2} + 1, & \text{if } \theta_{J-K-C} = 0 \\ x_{n-2}y_1 + x_{n-3}y_2 + \cdots + x_1y_{n-2}, & \text{if } \theta_{J-K-C} > 0 \end{cases} \quad (15)$$

where $\theta_{J-K-C} = x_{n-1}y_0 + x_{n-2}y_1 + \ldots + x_0y_{n-1}$. Note that the index θ_{J-K-C}, which is a function of input signals X and Y, determines the error compensation bias σ_{J-K-C}. Herein, we apply our generalized methodology to design other low-error fixed-width multipliers without sacrificing area. By applying the derivation result in Appendix A, one can recast (9) as

$$\sigma_{\text{Temp}} = \theta_{Q,w=0} + \left[\frac{1}{2}E_{\text{main}} - \theta_{Q,w=0} + \frac{1}{2}E_{\text{remain}}\right]_r = (\langle x_{n-2}y_1\rangle^{q_{n-2}} + \langle x_{n-3}y_2\rangle^{q_{n-3}} + \ldots + \langle x_1y_{n-2}\rangle^{q_1}) + [K]_r, \quad (16)$$

where

$$K = \langle x_{n-1}y_0\rangle^{q_{n-1}} + \langle x_0y_{n-1}\rangle^{q_0} + \frac{1}{2}E_{\text{main}} - \theta_{Q,w=0} + \frac{1}{2}E_{\text{remain}}. \quad (17)$$

In (16) and (17), since the elements of $\theta_{Q,w=0}$ except the first and last elements move to the n-th most significant column, we can flexibly control the compensation bias by input signals. In other words, since the generalized index $\theta_{Q,w=0}$ has different configurations corresponding to each value of Q, σ_{Temp} can provide distinct compensation biases. After applying binary thresholding scheme in the first step, we have proposed two types of

TABLE I
COMPARISON RESULTS OF RELATIVE MAXIMUM ERROR, $\varepsilon_{\max,\%}$

Multiplier	n=6	n=8	n=10	n=12	n=16
Direct-Truncated Structure	100% (321)	100% (1793)	100% (9217)	100% (45057)	100% (983040)
K-G-As' Structure	60.12%	71.44%	66.67%	72.73%	80.00%
J-K-Cs' Structure	33.33%	28.72%	26.07%	24.37%	22.31%
Type 2 with $Q=2^{n-1}+1, w=0$	27.73%	24.60%	22.84%	21.72%	20.37%
K-Ss' Structure with $w=1$	17.13%	14.67%	13.27%	12.37%	11.30%
Type 1 with $Q=0, w=1$	15.26%	13.22%	12.12%	11.43%	10.60%
K-Ss' Structure with $w=2$	12.77%	10.32%	8.95%	8.08%	7.04%
Type 1 with $Q=0, w=2$	11.53%	9.54%	8.36%	7.61%	6.69%

TABLE II
COMPARISON RESULTS OF RELATIVE AVERAGE ERROR, $\bar{\varepsilon}_{\%}$

Multiplier	n=6	n=8	n=10	n=12	n=16
Direct-Truncated Structure	100% (112.25)	100% (576.25)	100% (2816.35)	100% (13312.34)	100% (278525)
K-G-As' Structure	36.53%	32.68%	32.18%	29.32%	35.07%
J-K-Cs' Structure	33.20%	29.58%	26.16%	23.24%	18.88%
Type 2 with $Q=2^{n-1}+1, w=0$	21.44%	18.39%	16.20%	14.60%	12.41%
K-Ss' Structure with $w=1$	15.66%	12.83%	10.90%	9.55%	7.77%
Type 1 with $Q=0, w=1$	14.51%	12.00%	10.38%	9.23%	7.66%
K-Ss' Structure with $w=2$	14.35%	11.44%	9.49%	8.12%	6.33%
Type 1 with $Q=0, w=2$	14.08%	11.22%	9.34%	8.02%	6.30%

TABLE III
COMPARISON RESULTS OF RELATIVE VARIANCE OF ERROR, $\upsilon_{\%}$

Multiplier	n=6	n=8	n=10	n=12	n=16
Direct-Truncated Structure	100% (2222.03)	100% (54284.47)	100% (1172842)	100% (23650942)	100% (8561689532)
K-G-As' Structure	35.48%	42.29%	35.47%	40.05%	66.95%
J-K-Cs' Structure	24.20%	18.71%	16.27%	14.88%	13.11%
Type 2 with $Q=2^{n-1}+1, w=0$	14.43%	11.11%	9.56%	8.66%	7.65%
K-Ss' Structure with $w=1$	6.72%	4.79%	3.90%	3.39%	2.81%
Type 1 with $Q=0, w=1$	4.96%	3.80%	3.35%	3.09%	2.71%
K-Ss' Structure with $w=2$	4.71%	3.08%	2.37%	1.96%	1.51%
Type 1 with $Q=0, w=2$	4.28%	2.79%	2.18%	1.85%	1.47%

binary thresholding at $w = 0$ that have been clearly demonstrated in [8]. Through the full-search simulation for Type 2 binary thresholding in the second step, we observe that specific indexes $\theta_{Q=0,w=0}$, $\theta_{Q=1,w=0}$, and $\theta_{Q=2^{n-1},w=0}$ have almost the same error performance as that of $\theta_{Q=2^{n-1}+1,w=0}$. As shown in Tables I–III, Type 2 fixed-width multiplier with $\theta_{Q=2^{n-1}+1,w=0}$ achieves better performance than K-G-As' and J-K-Cs' structures. In Table IV, we can see that Type 2 fixed-width multiplier with $\theta_{Q=2^{n-1}+1,w=0}$ has the same active chip area as that of J-K-Cs' structure and possesses less than 13% area overhead compared with D-Truncated and K-G-As' structures. The resulting simple error-compensation biases with low truncation error in Type 2 binary thresholding are described as

$$\sigma_{\text{Type2,Q,w=0}} = \begin{cases} x_{n-2}y_1 + x_{n-3}y_2 + \cdots \\ \quad +x_1y_{n-2} + 1, & \text{if } \theta_{Q,w=0} < n \\ x_{n-2}y_1 + x_{n-3}y_2 + \cdots \\ \quad +x_1y_{n-2}, & \text{if } \theta_{Q,w=0} = n \end{cases} \tag{18}$$

where $\theta_{Q,w=0}$ could be one of the following: $\theta_{Q=0,w=0} = x_{n-1}y_0 + x_{n-2}y_1 + \ldots + x_0y_{n-1}$, $\theta_{Q=1,w=0} = x_{n-1}y_0 + x_{n-2}y_1 + \ldots + x_1y_{n-2} + \overline{x_0y_{n-1}}$, $\theta_{Q=2^{n-1},w=0} = \overline{x_{n-1}y_0} + x_{n-2}y_1 + \ldots + x_0y_{n-1}$, and $\theta_{Q=2^{n-1}+1,w=0} = \overline{x_{n-1}y_0} + x_{n-2}y_1 + \ldots + x_1y_{n-2} + \overline{x_0y_{n-1}}$. It is difficult to simulate that these indexes achieve better performance for $n > 16$ since the exhaustive simulation takes lots of computation time. In the third step, we show that these indexes are suitably adopted to design fixed-width multipliers for each value of n by statistical techniques. Because these statistical techniques at $w = 0$ have been fully exposed in [8, Sec. III], we bypass here. Throughout this paper, the probability of input bits is assumed to be uniform distribution, so we can approximate $(1/2)E_{\text{main}}$, $(1/2)E_{\text{remain}}$ and other terms with output expected values of logic-functions. Note that the error-compensation bias $\sigma_{\text{Type2,Q,w=0}}$ with $\theta_{Q=2^{n-1}+1,w=0}$ in (18) is the same as [8, eq. (23)] and other three multipliers are our new proposed structures in this paper. The corresponding low-error 8×8 fixed-width multipliers can be easily modified from the structure as shown in [8, Fig. 5].

TABLE IV
COMPARISON RESULTS OF CHIP-AREA RATIO IN PERCENTAGE, $R_{\%}$

Multiplier	n=6	n=8	n=10	n=12	n=16
Standard Structure	100% (62.06 um x 57.44 um)	100% (85.82 um x 82.64 um)	100% (108.92 um x 107.84 um)	100% (132.02 um x 128.00 um)	100% (178.22 um x 173.36 um)
Direct-Truncated Structure	43.51%	47.06%	46.08%	47.27%	49.13%
K-G-As' Structure	42.82%	46.52%	45.68%	46.93%	49.13%
J-K-Cs' Structure	55.66%	51.33%	52.77%	52.88%	53.33%
Type 2 with $Q=2^{n-1}+1, w=0$	55.66%	51.33%	52.77%	52.88%	53.33%
K-Ss' Structure with $w=1$	80.55%	74.30%	66.53%	64.25%	61.95%
Type 1 with $Q=0, w=1$	77.64%	67.25%	60.84%	62.65%	58.55%
K-Ss' Structure with $w=2$	94.68%	79.02%	72.15%	70.27%	66.96%
Type 1 with $Q=0, w=2$	97.87%	85.96%	74.52%	71.19%	67.58%

B. Realizable Error-Compensation Bias Keeping More Than n Most Significant Columns ($w \geq 1$)

In [1] and [2], the works turn out that lower truncation error can be obtained if larger $n+w$ most significant columns are kept in hardware. However, more area cost will be paid. Since the most significant columns of reduction error and rounding error do not possess the same column, we adopt S-Ss' method [2] to concurrently treat reduction and rounding errors. (9) can be rewritten as

$$\begin{aligned}\sigma_{\text{Temp}} &= \left\lfloor \frac{1}{2}E_{\text{main}} + \frac{1}{2}E_{\text{remain}} + \frac{1}{2^w}\theta_{Q,w} \right. \\ &\qquad \left. -E_{\text{reduct},w} + E_{\text{reduct},w} - \frac{1}{2^w}\theta_{Q,w} \right\rfloor_r \\ &\cong \left\lfloor \frac{1}{2}E_{\text{main}} + \frac{1}{2}E_{\text{remain}} + \frac{1}{2^w}\theta_{Q,w} - E_{\text{reduct},w} \right. \\ &\qquad \left. +E_{\text{reduct},w} - \frac{1}{2^w}\theta_{Q,w} + E_{\text{round},w} \right\rfloor \\ &\cong \left\lfloor \left(\frac{1}{2}E_{\text{main}} + \frac{1}{2}E_{\text{remain}} + \frac{1}{2^w}\theta_{Q,w} \right. \right. \\ &\qquad \left. \left. -E_{\text{reduct},w} \right) + \frac{[K]_r}{2^w} \right\rfloor \end{aligned} \tag{19}$$

where

$$K = 2^w\left(E_{\text{reduct},w} - \frac{1}{2^w}\theta_{Q,w} + E_{\text{round},w}\right) \tag{20}$$

$$\begin{aligned}E_{\text{reduct},w} &= \frac{1}{2^{w+1}}(x_{n-1-w}y_0 + x_{n-2-w}y_1 + \cdots \\ &\qquad + x_0 y_{n-1-w}) + \ldots + \frac{1}{2^n}x_0 y_0 \end{aligned} \tag{21}$$

$$E_{\text{round},w} = 2^{-1}(1-2^{-w}). \tag{22}$$

and where $\lfloor \bullet \rfloor$ denotes the maximum integer equal to or less than $\bullet$. Concurrently treating method for reduction and rounding errors of (19) can be directly referred to derivation of [2]. (22) can be found in [2], [6] and the only difference is the multiple of 2^n for fractional operation, so we ignore detailed derivations.

In the first step, two types of binary thresholding for the error-compensation bias can be changed to

Type 1)

$$\begin{aligned}&\sigma_{\text{Type1,Q,w}\geq 1} \\ &= \begin{cases} \left\lfloor \frac{1}{2}(E_{\text{main}} + E_{\text{remain}}) + \frac{1}{2^w}\theta_{Q,w} \right. \\ \left. -E_{\text{reduct},w} + \frac{[K_1]_r}{2^w} \right\rfloor, & \text{if } \theta_{Q,w} = 0 \\ \left\lfloor \frac{1}{2}(E_{\text{main}} + E_{\text{remain}}) + \frac{1}{2^w}\theta_{Q,w} \right. \\ \left. -E_{\text{reduct},w} + \frac{[K_2]_r}{2^w} \right\rfloor, & \text{if } \theta_{Q,w} > 0 \end{cases} \end{aligned} \tag{23}$$

Type 2)

$$\begin{aligned}&\sigma_{\text{Type2,Q,w}\geq 1} \\ &= \begin{cases} \left\lfloor \frac{1}{2}(E_{\text{main}} + E_{\text{remain}}) + \frac{1}{2^w}\theta_{Q,w} \right. \\ \left. -E_{\text{reduct},w} + \frac{[K_3]_r}{2^w} \right\rfloor, & \text{if } \theta_{Q,w} < n \\ \left\lfloor \frac{1}{2}(E_{\text{main}} + E_{\text{remain}}) + \frac{1}{2^w}\theta_{Q,w} \right. \\ \left. -E_{\text{reduct},w} + \frac{[K_4]_r}{2^w} \right\rfloor, & \text{if } \theta_{Q,w} = n \end{cases} \end{aligned} \tag{24}$$

where K_1, K_2, K_3 and K_4 are average values of K for satisfying $\theta_{Q,w\geq 1} = 0$, $\theta_{Q,w\geq 1} > 0$, $\theta_{Q,w\geq 1} < n$ and $\theta_{Q,w\geq 1} = n$, respectively. The restriction of the value of K can be limited as $[K_i]_r \in \{0, 1, 2^{w-1}-1, 2^{w-1}\}$ for $i = 1, 2, 3$ and 4. For $w \geq 1$, since the same simulation procedure as mentioned in [8] can be applied, we only introduce the analysis for $w = 1$ and construct the structure.

In the second step, by exhaustive search within Type 1 binary thresholding, we can find that one good index $\theta_{Q=0,w=1}$ for $n = 6$ satisfies the restriction of K_1 and K_2 as shown in Fig. 2. The average error and variance of error resulted from the index $\theta_{Q=0,w=1}$ are shown in Figs. 3 and 4, respectively. After exhaustive simulation from $n = 4$ to 16, we observe that the specific index $\theta_{Q=0,w=1}$ achieves better error performance, where the chosen index satisfies $[K_1]_r = 1$ and $[K_2]_r = 0$. On the other hand, for Type 2 binary thresholding, the error simulations in terms of average errors and variance of errors are larger than what we find error resulted from the best index in Type 1 binary thresholding, so we bypass the discussion in Type 2 binary

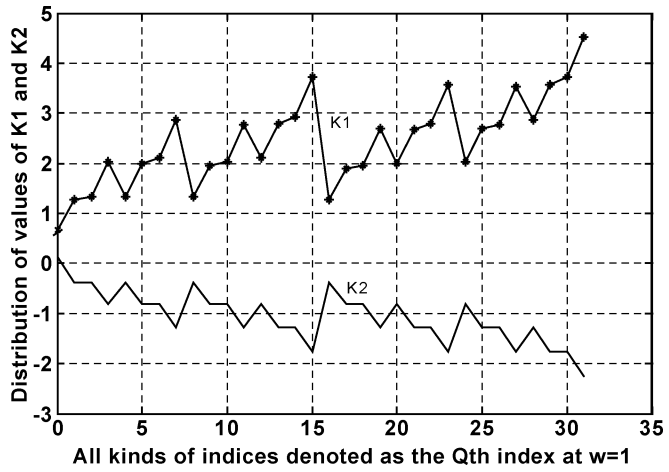

Fig. 2. Values of K_1 and K_2 versus different $\theta_{Q,w=1}$ in Type 1 binary thresholding for $n = 6$.

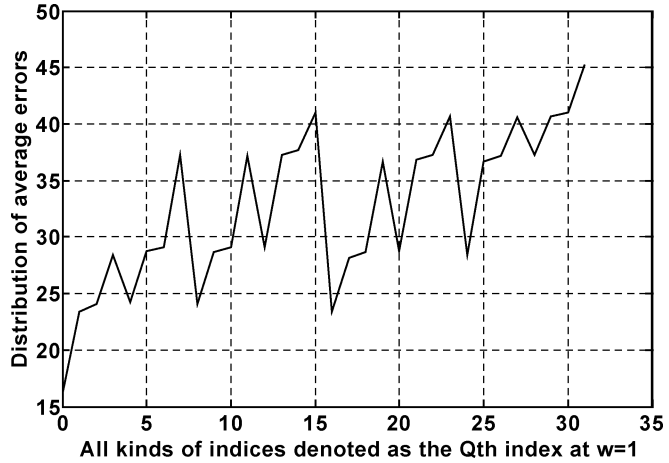

Fig. 3. Average errors by full-search simulation versus different $\theta_{Q,w=1}$ in Type 1 binary thresholding for $n = 6$.

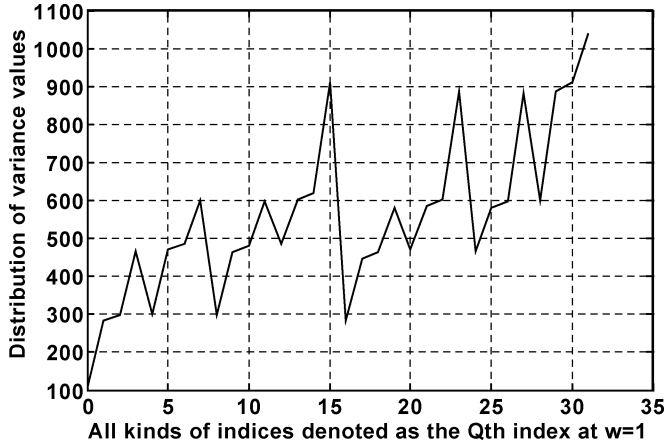

Fig. 4. Variance of errors by full-search simulation versus different $\theta_{Q,w=1}$ in Type 1 binary thresholding for $n = 6$.

thresholding. Thus, a new error-compensation bias at $w = 1$ can be proposed as

$$\sigma_{\text{Type1,Q=0,w=1}} = \begin{cases} \left\lfloor \frac{1}{2}(E_{\text{main}} + E_{\text{remain}} + \theta_{Q=0,w=1}) - E_{\text{reduct},w=1} + \frac{1}{2} \right\rfloor, & \text{if } \theta_{Q=0,w=1} = 0 \\ \left\lfloor \frac{1}{2}(E_{\text{main}} + E_{\text{remain}} + \theta_{Q=0,w=1}) - E_{\text{reduct},w=1} + 0 \right\rfloor, & \text{if } \theta_{Q=0,w=1} > 0. \end{cases} \tag{25}$$

Because $(1/2)E_{\text{remain}} = E_{\text{reduct},w=1}$, (25) can be simplified as

$$\sigma_{\text{Type1,Q=0,w=1}} = \begin{cases} \left\lfloor \frac{1}{2}(E_{\text{main}} + \theta_{Q=0,w=1}) + \frac{1}{2} \right\rfloor, & \text{if } \theta_{Q=0,w=1} = 0 \\ \left\lfloor \frac{1}{2}(E_{\text{main}} + \theta_{Q=0,w=1}) + 0 \right\rfloor, & \text{if } \theta_{Q=0,w=1} > 0 \end{cases} \tag{26}$$

where $\theta_{Q=0,w=1} = x_{n-2}y_0 + x_{n-3}y_1 + \ldots + x_0y_{n-2}$.

Importantly, the error performance obtained from the index $\theta_{Q=0,w=1}$ in Type 1 binary thresholding is lower than that of the existing structures [1]–[8] as shown in Tables I– III. In terms of area cost, the proposed one is competitive compared with K-Ss' structure with $w = 1$. In the third step, considering (25) with $\theta_{Q=0,w=1}$ in Type 1 binary thresholding, the statistical techniques based on the uniform distribution assumption can be used to verify values of K_1 and K_2. That is, we verify $[K_1]_r = 1$ and $[K_2]_r = 0$ at $w = 1$ in Type 1 binary thresholding for each value of n.

Case 1) $\theta_{Q=0,w=1} = 0$

In this case, $\theta_{Q=0,w=1} = 0$ is met only when $x_{n-2}y_0 = x_{n-3}y_1 = \ldots = x_0y_{n-2} = 0$. That is, Case 1 is a conditional probability case, and thus we deduce (27) from (20) as

$$\begin{aligned} K_1 &= E\left\{2\left(E_{\text{reduct},w=1} - \frac{1}{2}\theta_{Q=0,w=1} + E_{\text{round},w=1}\right)\right\} \\ &= 2 \times \left(\frac{1}{9} \times \left(\frac{1}{2^3}(n-2) + \frac{1}{2^4}(n-3) + \ldots + \frac{1}{2^n}\right) + \frac{1}{4}\right) \\ &\cong \frac{1}{18}(n-3) + \frac{1}{2}, \qquad \text{if } n \geq 4. \end{aligned} \tag{27}$$

Note that K_1 is positively proportional to $(1/18)n$, so this error-compensation circuit is difficult to design. Fortunately, while $n \geq 4$, we observe the fact that Case 1 is a minor case for all input combinations and the rounding value of K_1 is always equal to or greater than one. In order to design a simple error-compensation circuit based on above two facts, we can adopt a constant to approximate the value of K_1. (27) can be approximately set to $K_1 \cong 1$

$$K_1 \cong 1 \tag{28}$$

Substituting (28) into (23), we obtain

$$\sigma_{\text{Type1,Q=0,w=1}} = \left\lfloor \frac{1}{2}(E_{\text{main}} + E_{\text{remain}} + \theta_{Q=0,w=1}) - E_{\text{reduct},w=1} + \frac{1}{2} \right\rfloor, \quad \text{if } \theta_{Q=0,w=1} = 0. \tag{29}$$

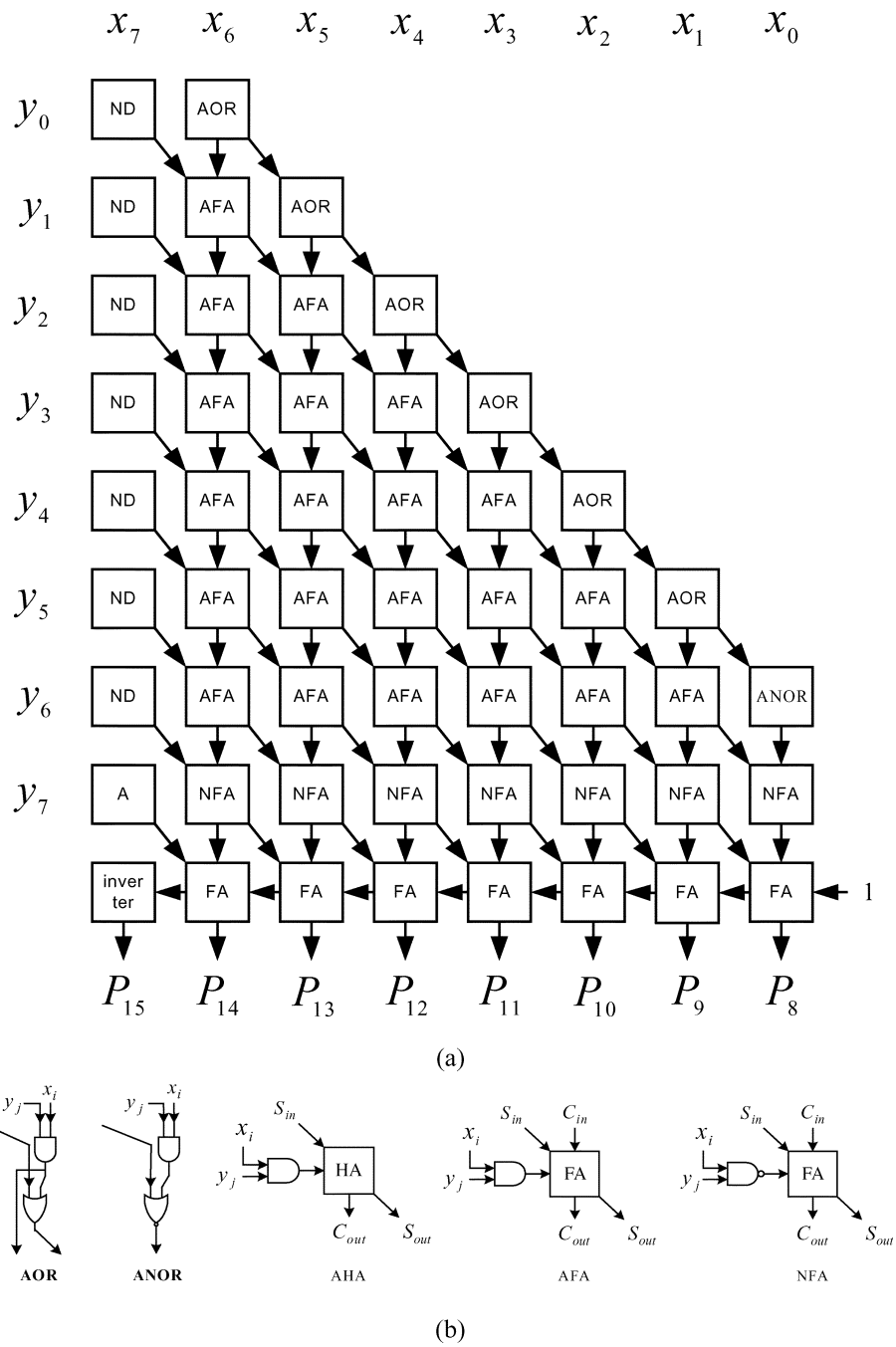

Fig. 5. (a) Proposed low-error fixed-width 8 × 8 two's-complement multiplier with $\theta_{Q=0,w=1}$, and (b) logic diagrams of AOR, ANOR, AHA, AFA, NFA.

Case 2) $\theta_{Q=0,w=1} > 0$

From (20), we obtain (30) by statistical techniques as

$$K_2 = E\left\{2\left(E_{\text{reduct},w=1} - \frac{1}{2}\theta_{Q=0,w=1} + 2^{-1}(1-2^{-1})\right)\right\}$$
$$= 2\times\left(-\frac{1}{8}(n-1)+\frac{1}{4} \times\left(\frac{1}{2^2}(n-1)+\frac{1}{2^3}(n-2)+\ldots+\frac{1}{2^n}\right)+\frac{1}{4}\right)$$
$$\cong 2\times\left(-\frac{1}{2^3}+\frac{1}{4}\right)=\frac{1}{4}, \qquad \text{if } n\geq 4. \tag{30}$$

Substituting (30) into (23), we obtain

$$\sigma_{\text{Type1,Q=0,w=1}} = \left\lfloor \frac{1}{2}(E_{\text{main}} + E_{\text{remain}} + \theta_{Q=0,w=1}) - E_{\text{reduct},w=1} + \frac{0}{2} \right\rfloor, \quad \text{if } \theta_{Q=0,w=1} > 0. \tag{31}$$

By combining (29) and (31), we obtain (25) and simplify it as (26). (26) with the chosen index $\theta_{Q=0,w=1}$ is suitable for being an error-compensation bias for each value of n. Importantly, note that if K_1 equals zero, (26) will be identical to the error-compensation bias in [4]–[6] for two's-complement multiplication.

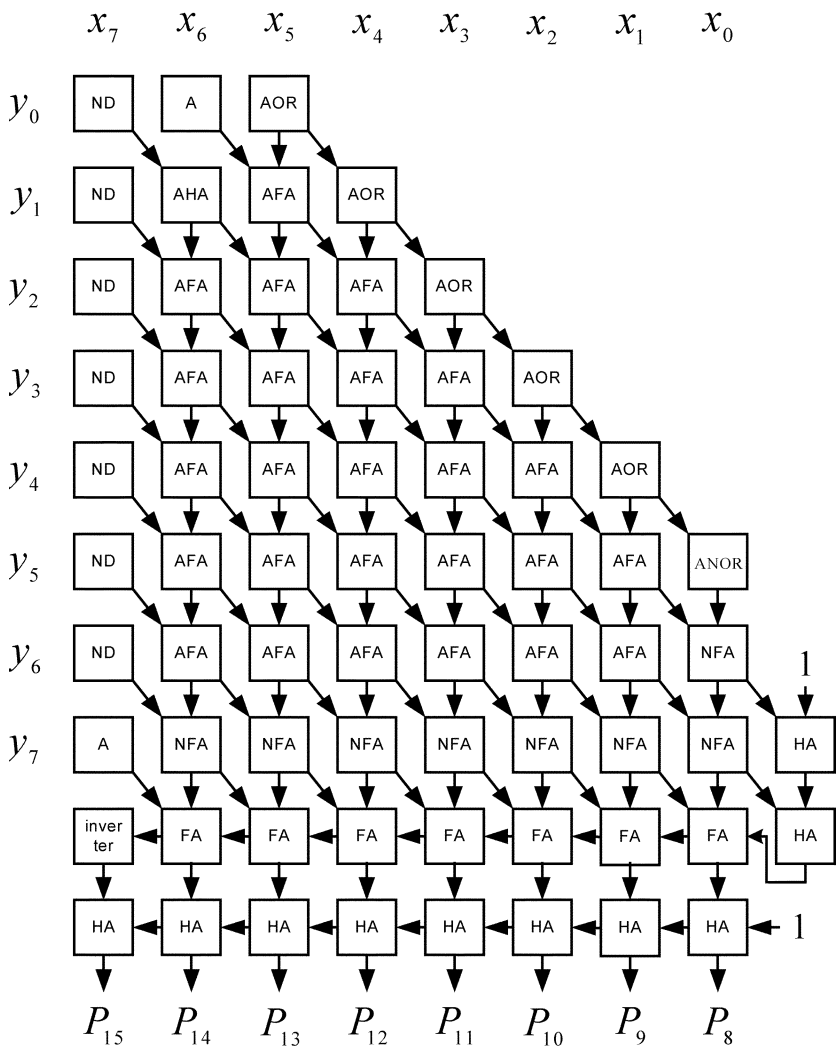

Fig. 6. Proposed low-error fixed-width 8 × 8 two's-complement multiplier with $\theta_{Q=0,w=2}$.

However, from simulation and statistical verification in (27), K_1 should be added a value that at least equal to or greater than one. Therefore, this constant approximation for K_1 can be accepted and the error can be lower than that in [4]–[6]. In the fourth step, (26) can be mapped to the structure as shown in Fig. 5(a) for $n = 8$, where A, ND, HA, and FA denote AND gate, NAND gate, a half adder and a full adder, respectively, and the logic diagrams of AOR, ANOR, AHA, AFA, NFA are depicted in Fig. 5(b). For other values of w, we can follow the same procedure to evaluate K, error performance and then construct the structure. From simulation results, $\theta_{Q=0,w}$ in Type 1 binary thresholding is still the best index for $w \geq 1$. Thus, we can conclude the error-compensation bias as

$$\sigma_{\text{Type1,Q=0,w}} = \begin{cases} \left\lfloor \frac{1}{2}(E_{\text{main}} + E_{\text{remain}}) + \frac{1}{2^w}\theta_{Q=0,w} - E_{\text{reduct},w} + \frac{1}{2} \right\rfloor, & \text{if } \theta_{Q=0,w} = 0 \\ \left\lfloor \frac{1}{2}(E_{\text{main}} + E_{\text{remain}}) + \frac{1}{2^w}\theta_{Q=0,w} - E_{\text{reduct},w} + \frac{1}{2} - \frac{1}{2^w} \right\rfloor, & \text{if } \theta_{Q=0,w} > 0. \end{cases} \quad (32)$$

The block diagram of the low-error fixed-width multiplier corresponding to (32) is shown in Fig. 6 for $w = 2$ and $n = 8$. Apparently, the area is increased due to requiring n half adders in the bottom row.

The proposed design procedure is summarized as follows.

1) Propose an error-compensation bias with a new binary thresholding for a fixed value of w.
2) Simulate the value of K and error performance of the proposed error-compensation bias using our generalized index, and then select the best index having lower error and satisfying the same value of K for limited width n.
3) Verify the realizable error compensation bias by statistical techniques.
4) Construct the low-error fixed-width multiplier structure.

We can see that utilizing our proposed methodology, not only the new multiplier is devised, but also the previous multiplier structures such as K-G-As', J-K-Cs', K-Ss' multipliers can be reproduced. So as to reproduce K-G-As' and K-Ss' multiplier, users only modify phase 1 design procedure as error-compensation bias with the same value of K (i.e., without binary thresholding) for a fixed value of w.

IV. Performance and Implementation

In this section, the performance is assessed in terms of the relative maximum error $\varepsilon_{\max,\%}$, relative average error $\overline{\varepsilon}_{\%}$, and relative variance of error $\upsilon_{\%}$ with respect to the error performance of the D-Truncated multiplier. The error terminologies are defined, respectively, as

$$\varepsilon_{\max,\%} = \frac{\varepsilon_{\max}}{\varepsilon_{\max(D-Truncated)}} \times 100\% \tag{33}$$

$$\overline{\varepsilon}_{\%} = \frac{\overline{\varepsilon}}{\overline{\varepsilon}_{\mathrm{D-Truncated}}} \times 100\% \tag{34}$$

$$\upsilon_{\%} = \frac{\upsilon}{\upsilon_{\mathrm{D-Truncated}}} \times 100\%. \tag{35}$$

It is obvious that a fixed-width multiplier is more accurate if $\varepsilon_{\max,\%}$, $\overline{\varepsilon}_{\%}$, and $\upsilon_{\%}$ are smaller. Tables I–III reveal simulated results corresponding to (33)–(35), respectively, for various fixed-width multipliers versus different width n. The comparison results show that our proposed fixed-width multipliers are more accurate than others [1]–[7] for different values of w. The better performance is achieved due to the fact that we derive better error-compensation biases to reduce the truncation error. Herein, note that in order to obtain more accurate multiplication result of K-Ss' structure, we adopt the rounding approach for n most significant columns. Thus, a slightly increased area can be expected for K-Ss' fixed-width multiplier.

Concerning the chip implementation, the cell-based design flow with Artisan standard cell library is adopted and the various fixed-width multipliers have been implemented in UMC 0.18-um CMOS process. The Synopsys Design Compiler is used to synthesize the RTL design of the conventional as well as the proposed multipliers and the Synopsys Apollo is adopted for placement and routing (P&R). The active chip area results are tabulated in Table IV and the chip-area ratio in percentage is defined as

$$R_{\%} = \frac{A_{\mathrm{Truncated}}}{A_{\mathrm{Standard}}} \times 100\% \tag{36}$$

where A_{Standard} and $A_{\mathrm{Truncated}}$ denote the area of the standard multiplier and various truncated multipliers, respectively. The chip-area ratio in Table IV shows that the proposed multiplier has approximately the same area-ratio of [4]–[7] at each corresponding value of w so that the low-error fixed-width multiplier is area-efficient. From Tables I–IV, it is concluded that if larger value of w is required, the features of lower error as well as larger area are exposed. For convenience of evaluating actual values of $\varepsilon_{\max}, \overline{\varepsilon}, \upsilon$, and $A_{\mathrm{Truncated}}$, we append the original error and area values of the D-Truncated multiplier and standard multiplier, respectively, to the second row within parentheses in Tables I–IV. On the other hand, more design information can be emerged, while the values captured from Tables I–IV are put together. For example, for $n = 8$, the relative average error and area ratio of Tables II and IV, respectively, can be plotted in Fig. 7, where X axis and Y axis denote area ratio and average error, respectively. In case the design constraint is under 70% area cost of the standard multiplier, we can see that, in Fig. 7, Type 1 structure with $\theta_{Q=0,w=1}$ is the best choice due to the lowest average error. In the same way, one can easily reproduce the similar figures for other values of n. The active chip layout area of the proposed Type 1 8 $\times$ 8 fixed-width multiplier as shown in Fig. 8 is 70.64 um $\times$ 67.52 um. In the worst case condition, the power consumption measured via the Synopsys Nanosim is 0.336 mW at an average operation rate of 100 MHz. The critical delay time obtained from the static timing analysis of Synopsys Apollo is 6.98 ns in the worst-case condition. Table V summarizes the chip characteristics of Type 1 8 $\times$ 8 fixed-width multiplier with $\theta_{Q=0,w=1}$. We find that, compared with the D-Truncated multiplier, our proposed Type 1 8 $\times$ 8 fixed-width multiplier can reduce 88% average error. It is also shown that the same proposed multiplier leads to 32.75% reduction in area compared with the standard multiplier, where the area of 8 $\times$ 8 standard multiplier is 85.82 um $\times$ 82.64 um. In the case of 16 $\times$ 16 multiplication, the product of Type 1 fixed-width multiplier with $\theta_{Q=0,w=1}$ is close to the product of the standard multiplier with an area reduction of 41.45%. Thus, our proposed fixed-width multipliers possess the features of low error and area efficiency.

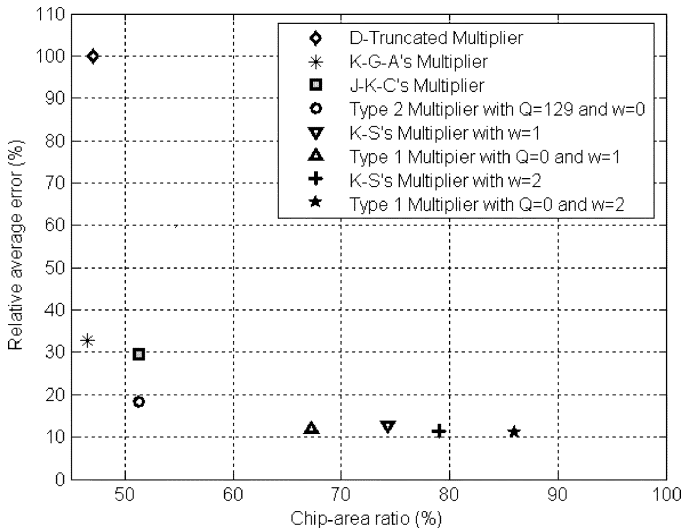

Fig. 7. Average error results versus area ratio for 8 $\times$ 8 fixed-width multipliers.

Next, we apply the proposed fixed-width multipliers to the 35-tap finite-impulse response digital filter for speech processing application. The detailed simulation setting can be referred to the literatures [8], [14]; here, we only expose the error performance of speech signals as shown in Fig. 9. From comparison results in Fig. 9 obtained by five fixed-width multipliers, we observed that the proposed Type 1 fixed-width multiplier with $\theta_{Q=0,w=1}$ outperform the existing fixed-width multipliers for most speech signals. Concerning the comparison of fixed-width multipliers at $w = 0$ (i.e., K-G-As', J-K-Cs' and Type 2 multipliers), more true voice can be retrieved after applying Type 2 fixed-width multiplier with $\theta_{Q=2^{n-1}+1,w=0}$. The two's-complement fractional multiplication including overflow detection and alleviation circuit design can be referred to the material in [15] and herein we bypass the discussion.

V. Conclusion

This paper develops a more generalized methodology for designing a family of low-error area-efficient fixed-width multipliers for different values of w. In this article, by properly choosing the binary thresholding and generalized indexes as mentioned in this methodology, we devise five new better error-

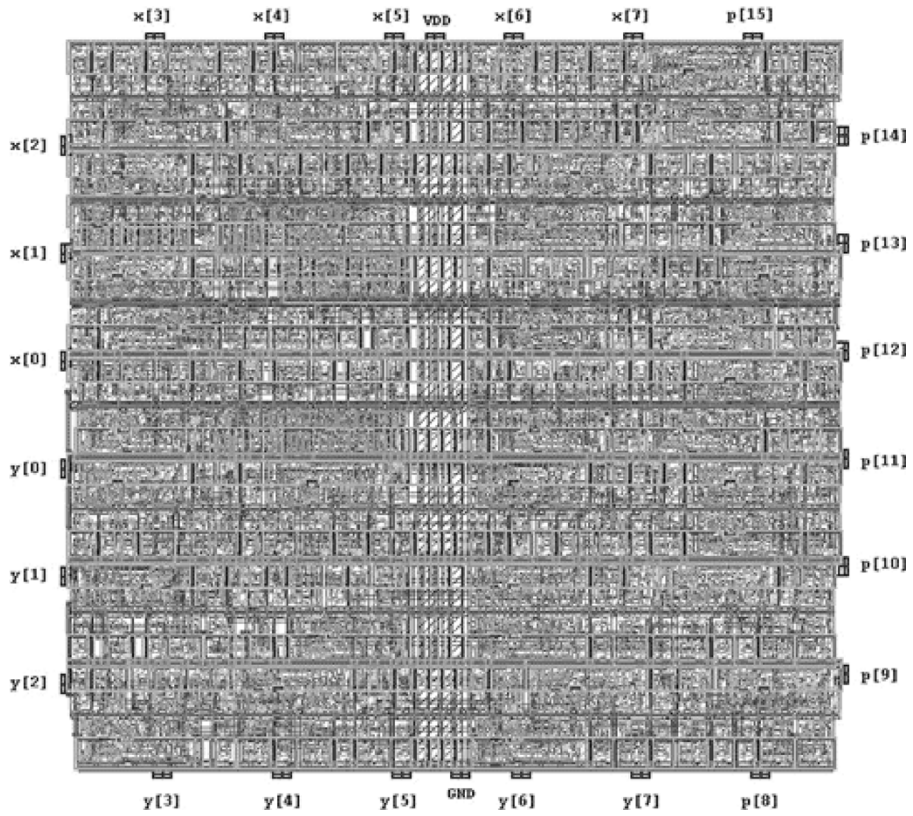

Fig. 8. Type 1 fixed-width multiplier layout with $\theta_{Q=0,w=1}$ for $n = 8$.

TABLE V
CHIP CHARACTERISTICS OF TYPE 1 8 × 8 FIXED-WIDTH MULTIPLIER WITH $\theta_{Q=0,w=1}$

Multiplier & Multiplicand Word Length (n)	8 bits
Product Word Length (n)	8 bits
Critical Delay Time	6.98 ns
Power Supply	1.8 V
Power Consumption @ 100MHz	0.336 mW
Active Chip Area	70.64um x 67.52um
Process Technology	UMC 0.18 um CMOS

compensation biases to reduce the relative maximum error, relative average error and relative variance of error. From chip-area ratio comparison, these new low-error fixed-width multipliers are shown to be area-efficient for VLSI implementation. Finally, we successfully apply the proposed fixed-width multipliers to speech processing application and obtain satisfactory results. The future work is to apply this generalized methodology to other arithmetic number systems and we have some comparable results for fixed-width Booth multipliers [16], [17].

APPENDIX A

By introducing an integer variable, $\theta_{Q,w=0}$, into (9), we can obtain

$$\sigma_{\text{Temp}} = \left[\frac{1}{2}E_{\text{main}} + \theta_{Q,w=0} - \theta_{Q,w=0} + \frac{1}{2}E_{\text{remain}}\right]_r. \tag{A.1}$$

Let

$$A = \theta_{Q,w=0} \tag{A.2}$$

$$B = \frac{1}{2}E_{\text{main}} - \theta_{Q,w=0} + \frac{1}{2}E_{\text{remain}} \tag{A.3}$$

$$|B| = \Phi + \phi \tag{A.4}$$

where A is a nonnegative integer since $\theta_{Q,w=0}$ is a nonnegative integer, B is a real number, Φ is a positive integer and ϕ is a fractional number whose range is $0 \le \phi < 1$. In view of (A. 2) to (A.4), there are only two conditions to be discussed in the sequel.

Case 1) $A \ge 0$ and $B \ge 0$

Since $A \ge 0$ and $B \ge 0$, (A.5) is directly obtained as

$$[A + B]_r = A + [B]_r. \tag{A.5}$$

Case 2) $A > 0$ and $B < 0$

$$\begin{aligned}[A + B]_r &= [A - |B|]_r \\ &= [A - \Phi - \phi]_r.\end{aligned} \tag{A.6}$$

Applying Lemma 1 in which $\Psi = A - \Phi$ and $\psi = \phi$, (A.6) can be written as

$$A - \Phi - [\phi]_r = A - (\Phi + [\phi]_r) = A - [\Phi + \phi]_r = A - [B]_r. \tag{A.7}$$

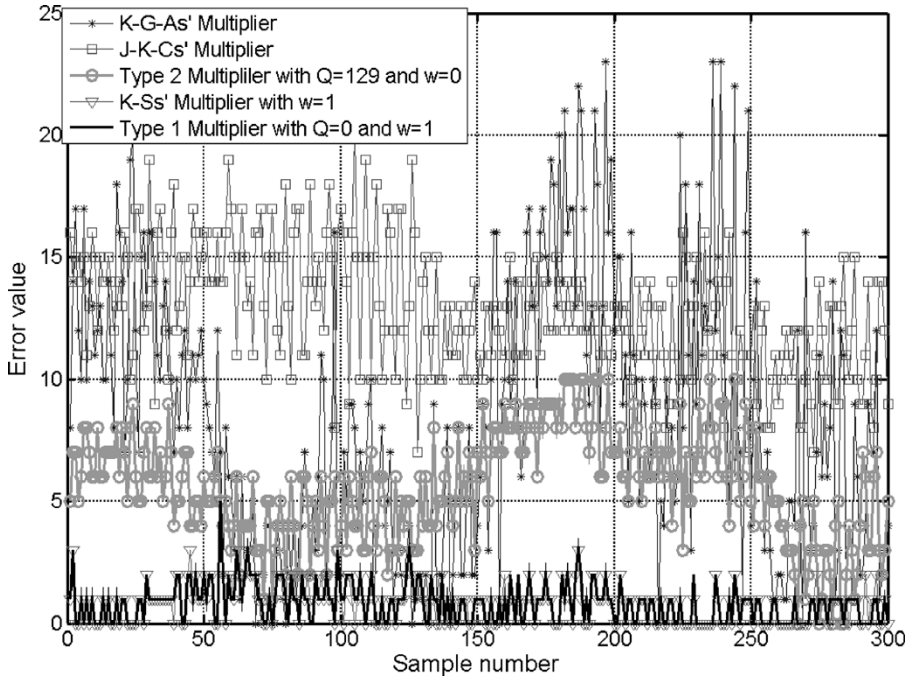

Fig. 9. Comparison results of error signals obtained with five kinds of fixed-width multipliers.

Combining (A.5) to (A.7), we conclude $[A+B]_r = A + [B]_r$ and

$$\sigma_{\text{Temp}} = \theta_{Q,w=0} + \left[\frac{1}{2}E_{\text{main}} - \theta_{Q,w=0} + \frac{1}{2}E_{\text{remain}}\right]_r. \tag{A.8}$$

Lemma 1: If given an integer variable, Ψ, and fractional number, ψ, whose range is $0 \leq \psi < 1$, then $[\Psi - \psi]_r = \Psi - [\psi]_r$.

Proof: If $\Psi \geq 1$, we directly obtain the result as

$$[\Psi - \psi]_r = \Psi - [\psi]_r. \tag{A.9}$$

If $\Psi \leq 0$, we can modify $[\Psi - \psi]_r$ as

$$[\Psi - \psi]_r = -[-\Psi + \psi]_r = -(-\Psi + [\psi]_r) = \Psi - [\psi]_r. \tag{A.10}$$

Combining (A.9) and (A.10), we have

$$[\Psi - \psi]_r = \Psi - [\psi]_r.$$

□

ACKNOWLEDGMENT

The authors would like to thank the anonymous referees for their valuable suggestions to this paper.

REFERENCES

[1] Y. C. Lim, "Single-precision multiplier with reduced circuit complexity for signal processing applications," *IEEE Trans. Comput.*, vol. 41, no. 10, pp. 1333–1336, Oct. 1992.

[2] M. J. Schulte and E. E. Swartzlander Jr., "Truncated multiplication with correction constant," in *VLSI Signal Processing, VI*. New York: IEEE Press, 1993, pp. 388–396.

[3] S. S. Kidambi, F. El-Guibaly, and A. Antoniou, "Area-efficient multipliers for digital signal processing applications," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal. Process.*, vol. 43, no. 2, pp. 90–94, Feb. 1996.

[4] E. J. King and E. E. Swartzlander Jr., "Data-dependent truncation scheme for parallel multipliers," in *Proc. 31st Asilomar Conf. Signals, Systems, and Computers*, vol. 2, Pacific Grove, CA, 1997, pp. 1178–1182.

[5] E. E. Swartzlander Jr., "Truncated multiplication with approximate rounding," in *Proc. 33rd Asilomar Conf. Signals, Systems, and Computers*, vol. 2, 1999, pp. 1480–1483.

[6] M. J. Schulte, J. E. Stine, and J. G. Jansen, "Reduced power dissipation through truncated multiplication," in *Proc. IEEE Alessandro Volta Memorial Workshop on Low Power Design*, Como, Italy, 1999, pp. 61–69.

[7] J. M. Jou, S. R. Kuang, and R. D. Chen, "Design of low-error fixed-width multiplier for DSP applications," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal. Process.*, vol. 46, no. 6, pp. 836–842, Jun. 1999.

[8] L. D. Van, S. S. Wang, and W. S. Feng, "Design of the lower-error fixed-width multiplier and its application," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal. Process.*, vol. 47, pp. 1112–1118, Oct. 2000.

[9] S. L. Freeny, "Special-purpose hardware for digital filtering," *Proc. IEEE*, vol. 63, no. 4, pp. 633–647, Apr. 1975.

[10] L. D. Van, "A new 2-D systolic digital filter architecture without global broadcast," *IEEE Trans. Very Large-Scale Integr. (VLSI) Syst.*, vol. 10, pp. 477–486, Aug. 2002.

[11] C. R. Baugh and B. A. Wooley, "A two's complement parallel array multiplication algorithm," *IEEE Trans. Comp.*, vol. C-22, no. 12, pp. 1045–1047, Dec. 1973.

[12] K. Hwang, *Computer Arithmetic: Principles, Architecture, and Design*. New York: Wiley, 1979.

[13] F. Cavanagh, *Digital Computer Arithmetic: Design and Implementation*. New York: McGraw-Hill, 1984.

[14] M. E. Paul and K. Bruce, *C Language Algorithms for Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[15] L. D. Van, "Design of efficient VLSI architectures: Multiplier 2-D digital filter, and adaptive digital filter," Ph.D. dissertation, Dept. Elect. Eng., National Taiwan University, Taipei, Taiwan, R.O.C, 2001.

[16] M. A. Song, L. D. Van, T. C. Huang, and S. Y. Kuo, "A generalized methodology for low-error and area-time efficient fixed-width Booth multipliers," in *Proc. IEEE Int. Midwest Symp. Circuits Systems*, vol. 1, Japan, Jul. 2004, pp. 9–12.

[17] M. A. Song, L. D. Van, C. C. Yang, S. C. Chiu, and S. Y. Kuo, "A framework for the design of error-aware power-efficient fixed-width Booth multipliers," in *Proc. IEEE Int. Symp. Circuits Systems*, Kobe, Japan, May 2005.

Lan-Da Van (S'98–M'02) was born in Miaoli, Taiwan, R.O.C., in, 1972. He received the B.S. (Honors) and the M.S. degree from Tatung Institute of Technology, Taipei, Taiwan, R.O.C. in 1995 and 1997, respectively, and the Ph.D. degree from National Taiwan University (NTU), Taipei, Taiwan, R.O.C. in 2001, all in electrical engineering.

From 1997 to 2001, he was a Research Assistant at National Taiwan University. Since 2001, he is a Researcher at Chip Implementation Center (CIC), National Applied Research Laboratories, Hsinchu, Taiwan, R.O.C. His current research interests are in very lareg-scale integration architectures, algorithms, and chips for digital signal processing (DSP), baseband communication systems, and computer applications. This includes the design of high-performance/low-power/area-aware DSP processors, adaptive digital filters, transform, computer arithmetic, and platform-based system-on-a-chip (SOC) designs. He has published 20 IEEE journal and conference papers in these areas.

Dr. Van was a recipient of the Chunghwa Picture Tube (CPT) and Motorola fellowships in 1996 and 1997, respectively. He was an elected chairman of IEEE NTU Student Branch in 2000. In 2002, he has received IEEE award for outstanding leadership and service to the IEEE NTU Student Branch. In 2003 and 2004, he has been listed in Marquis Who's Who in Science and Engineering. In 2005, he was a recipient of the Best Poster Award at iNEER Conference for Engineering Education and Research (*iCEER*). Presently, he serves as a reviewer for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: FUNDAMENTAL THEORY AND APPLICATIONS, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: ANALOG AND DIGITAL SIGNAL PROCESSING, IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE SIGNAL PROCESSING LETTERS, *Elsevier Microelectronics Journal*, *Elsevier Integration*, *The VLSI Journal*, and *EURASIP Journal on Applied Signal Processing*.

Chih-Chyau Yang was born in Changhua, Taiwan, R.O.C., in 1974. He received the B.S. degree in electrical engineering from National Cheng Kung University, Tainan, Taiwan, R.O.C. in 1986, and the M.S. degree in electronics engineering from National Chiao Tung University, Hsinchu, Taiwan, R.O.C. in 1989.

He is currently with the Chip Implementation Center (CIC), National Applied Research Laboratories, Hsinchu, Taiwan, R.O.C. His research interests include very large scale integration signal processing, processor architecture, and platform-based system-on-a-chip (SOC) designs.