Abstract—This brief proposes a dynamic error-compensation circuit for a fixed-width squarer based on the Booth folding technique. According to the expected value of the partial product through the Booth encoder, a closed form of the compensated value can be derived, including column information that can be used to improve accuracy. The proposed compensation circuit was derived using a mathematical probability model, which means that it is easily implemented for bit-lengths of 32, 64, and longer. Implemented using the TSMC 0.18-μm CMOS process, the proposed 32-bit squarer achieved an operation frequency of 50 MHz and gate count of 3.7 k. Compared with previous solutions, the proposed squarer achieves the best tradeoff between area-efficiency, cost, and accuracy.

Index Terms—Fixed-width squarer, Booth folding technique, probability theory, dynamic error-compensation

I. INTRODUCTION

Fixed-width squaring is a function widely used in digital signal processing (DSP) applications, such as image compression, pattern recognition, adaptive filtering, vector quantization, and equalization [1]. In DSP applications, the ability to generate outputs with a width the same as the input is critical. Thus, many researchers have focused on the development of fixed-width squarers to reduce area overhead and minimize quantization error [2]-[6]. The most accurate approach is the post truncation (P-T) squarer, which saves established time in the compensating circuit, and a conditional probability estimator (ACPE) has also been introduced for Booth multipliers [12], which saves time in the compensating circuit. To reduce area overhead, the truncated fixed-width squarers with error-compensation circuits for simple folding [4]-[5] and BFT [6] techniques are presented. The merged partial products squarer with an error-compensation circuit proposed in [4] has proved a highly efficient design with regard to power usage and area efficiency. Linear compensation was proposed with the aim of improving accuracy [5] and [6] presented two error-compensated methods based on two truncation bit groups. The use of BFT has been shown to enhance accuracy by reducing the number of truncated partial products. Thus, this study aims to design an error-compensation circuit in truncated squarer by using BFT architecture.

This paper proposes an area-efficient fixed-width squarer, referred to as the probability Booth-folding technique (PBFT). We adopted BFT to reduce the number of partial products and introduced a uniform equation for error-compensation bias based on probability theory. The proposed PBFT fixed-width squarer is systematically designed and results in high accuracy and reasonable area overhead.

This paper is organized as follows. In Section II, we present the background of BFT. In Section III, we outline the proposed PBFT algorithm and discuss the circuit in Section IV. Section V compares the proposed method with previous approaches with regard to error performance and area overhead. Conclusions are drawn in Section VI.

II. BOOTH FOLDING TECHNIQUE

L-bit numbers X and its $2L$-bit standard squaring product $P$ can be expressed in two’s complement representation as follows:

$$
P = X^2
$$

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

Partial products can be efficiently reduced using the Booth-folding technique. Based on the Booth encoder [14], three consecutive numbers in $X$ are transformed into $y_i$, where

$$
y_i = -2 \cdot x_{2i+1} + x_{2i} + x_{2i-1} \text{ (assuming } x_{-1} = 0) \text{ and } y_i \in \{0, \pm1, \pm2\}.$$

Table I shows the mapping table of the
The MP includes the most significant products, as shown in Fig. 2, and products can be obtained from Table II, in which takes fixed-width squarer can be approximated from (5):

\[ P_q \approx MP + TP \]  \hspace{1cm} (6)

\[ TP = \left[ TP_{mj} + \sigma \cdot 2^{L-w} \right] \]  \hspace{1cm} (7)

\[ \sigma = \text{Round} \left( TP_{mi} + TP_{rd} \right) \]  \hspace{1cm} (8)

where \( \lfloor \cdot \rfloor \) and \( \text{Round} (\cdot) \) indicate the floor and rounding operator, respectively. The \( TP_{mj} \) is the major section of TP that contains \( w \) columns of TP and uses real information for the calculation of TP. The \( TP_{mi} \) is the minor section of TP, and the expected values are used to estimate \( \sigma \). The \( TP_{rd} \) is compensated bias for rounding error. \( TP_{mj}, TP_{mi}, \) and \( TP_{rd} \) are then defined as follows:

\[ TP_{mj} = T_{c_1} + T_{c_2} + \cdots + T_{cw} \]  \hspace{1cm} (9)

\[ TP_{mi} = T_{r_1} + T_{r_2} + \cdots + T_{r^\beta} \]  \hspace{1cm} (10)

\[ TP_{rd} = \left( \frac{1}{2} \sum_{i=L-w}^{L-1} 2^i \right) \cdot 2^{-\left(L-w \right)} \]  \hspace{1cm} (11)

where \( \beta = \left[ \frac{L-w}{2} \right] \) and \( \lceil \cdot \rceil \) indicates the ceiling operator. \( T_{c_1} \ldots T_{c_L} \) and \( T_{r_1} \ldots T_{r^\beta} \) are the row block and column block of TP, respectively. Table III shows the expected value of partial products in \( TP_{mi} \), which are derived from Table I and Table II. Taking \( E[p_{2,3}] \) and \( E[c_{1,2}] \) as examples, the expected values can be obtained as follows:

\[ E[p_{2,3}] = \sum_{k=0,1,\pm 2} P[p_{2,3}] = 1|y_2 = k|P[y_2 = k] \]

\[ = 0 - \frac{1}{4} + \frac{1}{2} - \frac{1}{4} + \frac{1}{2} - \frac{1}{4} + \frac{1}{2} - \frac{1}{8} + \frac{1}{2} - \frac{1}{8} \]

\[ = \frac{3}{8} \]  \hspace{1cm} (12)
where $P[\cdot]$ and $E[\cdot]$ are the probability and expected operators. The expected distribution of $TP_{mi}$ is summarized in Fig. 3.

**B. Proposed Compensation Method**

By substituting partial products into (10) with the expected values from Table III and Fig. 3, the expected value for $TP_{mi}$ can be expressed as follows:

$$
E[T_{r1}] = A(L-w) + B(L-w)
$$
$$
E[T_{r2}] = A(L-w-4) + B(L-w-4)
$$
$$
\vdots
$$
$$
E[T_{ri}] = A(L-w-(i-1)4) + B(L-w-(i-1)4)
$$
$$
\vdots
$$
$$
E[T_{r\beta}] = \theta_w
$$

where $A(\cdot)$, $B(\cdot)$, and $\theta_w$ are represented by the deep, light, and special regions in Fig. 3 as follows:

$$
A(n) = \frac{3}{8} \left( 1 - \frac{1}{2^n} \right)
$$

$$
B(n) = \frac{7}{32} \times \frac{1}{2^n}
$$

$$
\theta_w = \begin{cases} 
L-w = 4m & : \frac{1}{12} \\
L-w = 4m+1 & : \frac{7}{16} \\
L-w = 4m+2 & : \frac{3}{4} \\
L-w = 4m+3 & : \frac{3}{2} 
\end{cases}
$$

Based on (11) and (18), a constant value can be pre-calculated, such that the compensation circuit sums $TP_{mj}$ and $\sigma$ with full adders (FAs) and half adders (HAs) using the CSA architecture, thereby ensuring high-speed computation. The CSA array then sums MP and the carries from compensated circuit.

**IV. PROPOSED SQUARER ARCHITECTURE**

According to (7) and (20), truncated bias can be estimated according to the word-length and columns of $TP_{mj}$. Figure 4 shows the entire architecture of the proposed low-error squarer, comprising a Booth encoder, carry-save architecture (CSA) array, carry propagation adder (CPA), and compensation circuit. Based on (11) and (18), a constant value can be pre-calculated, such that the compensation circuit sums $TP_{mj}$ and $\sigma$ with full adders (FAs) and half adders (HAs) using the CSA architecture, thereby ensuring high-speed computation.
Finally CPA produces the results. The CPA is implemented as the high speed Brent-Kung parallel-prefix adder. The following steps illustrate the design flow of the proposed squarer.

1) Use word-length $L$ and column information $w$ to calculate $\beta$ and $\sigma$ using (10) and (19), respectively.
2) Calculate the constant value for $\sigma$ from (20) and TP by summing $TP_{m,j}$ and $\sigma$ using (7).
3) Use the CSA structure to sum MP and the corresponding carries, whereupon the CPA produces the final products.

Figure 5 shows an example of the compensation circuit of proposed PBFT squarer ($L = 16$ and $w = 3$).

V. COMPARISON AND DISCUSSION

A. Accuracy

Accuracy is an important metric in the design of fixed-width squarers. This paper uses signal-to-noise ratio (SNR) as a means of estimating accuracy. The definition of SNR is as follows:

$$SNR [dB] = 10 \log \left( \frac{E[P^2]}{E[(P - P_q)^2]} \right).$$

Table IV presents a comparison of the proposed fixed-width squarer and previous methods with regard to SNR. The SNR values are obtained based on the full range input pattern (exhaustive input). The most accurate scheme is the P-T squarer, which calculates all the partial products and outputs the least significant bit (LSB) via a rounding operation. Thus, all compensation methods aim to match the accuracy of P-T squarer. The D-T squarer also achieved higher accuracy than the previous fixed-width Booth multipliers. In addition, the proposed PBFT squarer achieved higher accuracy than the previous fixed-width Booth multipliers produces more partial products in TP compared to the BFT fixed-width squarer. However, the proposed PBFT squarer achieved higher accuracy than the fixed-width Booth multiplier. In addition, the proposed PBFT squarer also achieved higher accuracy than the previous fixed-width squarers in [3] and [6]-[5].

![Fig. 5. Example of 16-bit $w = 3$ compensation circuit.](image)

![Fig. 6. Comparisons of area-efficiency between the proposed and the existing squarers.](image)
external data was serially input to and output by the TM, which fed input data to the squarer in parallel, and captured the paralleled output data in function mode.

D. System Application

The proposed PBFT squarer is then applied to a pseudo random number generator (PRNG) [15] for demonstrating the performance. The PRNG executes a 32-bit nonlinear equation \( X_t = \alpha X_{t-1} + \beta X_{t-2} \), thus, a 32-bit squarer is implemented in PRNG circuit. By using TSMC 40-nm standard cell library to synthesize the PRNG, the PRNG adopting PBFT squarer can achieve a high through rate of 12.8 Gb/s with 4,859 \( \mu \text{m}^2 \) area cost, which is saved 25% circuit area when compared with the P-T squarer application.

VI. CONCLUSION

This paper proposes a simple, uniform formula with which to design compensated fixed-width squarers. The proposed PBFT algorithm is based on the probability theory of BFT, the efficiency of which enables its application with lengthy squarers to achieve high accuracy. The proposed method is also cost effective due to the simplicity of the compensation method. Among the methods compared, the proposed PBFT squarer provides the best tradeoff between accuracy, area overhead, and cost performance.

**TABLE IV**

**Comparisons of accuracy, area, and delay between the proposed and existing methods**

<table>
<thead>
<tr>
<th>L</th>
<th>Methods</th>
<th>( L = 8 )</th>
<th>( L = 10 )</th>
<th>( L = 12 )</th>
<th>( L = 14 )</th>
<th>( L = 16 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>P-T [1]</td>
<td>46.4</td>
<td>48.0</td>
<td>50.3</td>
<td>53.0</td>
<td>56.3</td>
</tr>
<tr>
<td>2</td>
<td>PBT [2]</td>
<td>48.3</td>
<td>50.0</td>
<td>52.3</td>
<td>54.9</td>
<td>57.0</td>
</tr>
<tr>
<td>3</td>
<td>Proposed</td>
<td>48.3</td>
<td>50.0</td>
<td>52.3</td>
<td>54.9</td>
<td>57.0</td>
</tr>
</tbody>
</table>

*: Fixed-width multipliers.

**Fig. 7.** Chip photomicrograph and characteristics of proposed 32-bit PBFT squarer.

**REFERENCES**


