Comparison of Multiplexer based Pseudo-Carry Compensation Truncated Multiplier with Standard Multipliers

Sunil. M1 Abhilash K.G2 Harish .B3
1,2,3PG Student 3Assistant Professor
1,2,3Department of Electronics Communication Engineering
1,2,3SIET, Tumkur, Karnataka, India

Abstract— The conventional multipliers consume more power and area and causes more delay in the system. To overcome these problems, truncated multipliers is applied where it consumes less area, power and delay. But in Truncated multiplier probability of error is more due to truncation of LSP’s, hence compensation circuit need to be added to overcome the probability of error. In this project work Pseudo-carry compensation truncation scheme is implemented where compensation circuit is added to regular truncated multipliers. The proposed PCT multiplier is compared with the conventional multipliers like Braun multipliers, Vedic multipliers, and Booth multiplier and the result is analyzed. The PCT multiplier can be used in the field of Image processing, Digital signal processing Digital communication etc.

Key words: PCT Truncated Multiplier, FPGA, VLSI Design

I. INTRODUCTION

Truncated multiplier is one of the application related multiplier method. It is widely used in DSP (Digital signal processing) application, like FIR (finite impulse response) and DCT (discrete cosine transform), IIR (Infinite impulse response). In truncated multiplier, design complexity is reduced as compared to the conventional designs. The only difference between this multiplier with conventional designs is that, truncated multiplier is an application based method, which is used where exactly fixed result is required, it computes only ‘n’ bits output this is due to the truncating the LSP bits. Whereas output of the conventional designs is ‘2n’ bits. By truncating LSP bits power and area is reduced significantly. The architecture of truncated multiplier requires FA and HA (full adder and half adder) and it also need a multiplexer (4:1 multiplexer) and finally it requires the RCA (ripple carry adder) block to add the final result.

II. PCT TRUNCATED MULTIPLIER

The Fig 1 shows the block diagram of pseudo-carry compensation truncated multiplier. This type of multiplication method is designed using two type adders, full adders and half adders with 4:1 multiplexers then it gives FMUX and HMUX. As shown in Fig 1 FMUX are placed at the interior of the design and HMUX is placed at near to the correction block. A fixed correction bias of ‘1’ is used to compensate the error. And it used as input to the HMUX. The RCA (ripple carry adder) is used at the bottom of the architecture to add the generated partial product bit from the FMUX block to get the final product. Instead of ripple carry adder (RCA), logical OR gates are used to add the partial product. But if OR gates are used area consumption is the major problem, such RCA block is the better choice. Hence PCT truncated multiplier widely used in DSP processors and also it is used in VLSI designs.
Comparison of Multiplexer based Pseudo-Carry Compensation Truncated Multiplier with Standard Multipliers

Finally, the $n$-bit truncated product, $P_t$, with its least significant bit weighted $2^0$, is given by

$$P = \sum_{i=r}^{n-1} M_i 2^{i-n} + \sum_{i=r+1/2}^{n-1} M_{ti} 2^{i-n} + \sum_{i=r+1/2}^{n-1} (X_i \cdot Y_i) 2^{i-n} + \sum_{i=0}^{r-1} (X_i \cdot Y_i) 2^{i-n} + 2^{-1} \quad (2.3)$$

Fig. 3: RTL View of PCT Truncated Multiplier

The figure 2 shows RTL view of a PCT Truncated multiplier matrix multiplier for an $8 \times 8$-bit multiplier; it is again an array multiplier.

III. PCT COMPARISON WITH STANDARD MULTIPLIERS

Proposed standard multipliers are designed using verilog with 64 bit and implemented on Field Programmable Gate Array (FPGA) devices. The main important proposed multipliers are Braun multiplier, Vedic multiplier (Urdhva Triyakbhyam Sutra, Nikhilam Sutra), Truncated multiplier.

A. Braun Multiplier:

Braun multiplier is also called as Carry save array multiplier, this is because the architecture of Braun multiplier is similar to the parallel multiplier, carry from the previous bit is saved. The architecture of Braun multiplier consists of array of AND gates and adders, and it doesn’t requires logic registers. For ‘$n \times n$’ bit, Braun architecture require ‘$(n-1)$’ adders and ‘$n2$’ AND gates. The architecture of Braun multiplier is illustrated in Fig 4, hence ‘$4 \times 4$’ bit Braun architecture requires 12 adders and 16 AND gates as illustrated in Fig 4.

B. Vedic Multiplier:

Vedic multiplier is contains 16 main sutras and 16 sub sutras and these each sutra has an different architecture and the power consumption, delay, and area utilization of these sutras are different. The main important sutras in vedic multipliers are Urdhva Triyakbhyam Sutra, Nikhilam Sutra.

1) Urdhva Triyakbhyam Sutra:

Urdhva Triyakbhyam sutra (algorithm) is one of the effective method in Vedic sutras, the delay in this sutra is very less compare to the other sutras. In earlier days this sutra is used to multiply the two decimal numbers. Later this sutra is applied to both binary and decimal numbers.
Consider the multiplication of two decimal numbers multiplicand (43) X multiplier (68), the digit on the both the sides are multiplied and the carry from the previous step is added to it. The example of this method is illustrated in Fig 7. If vertical cross lines are more than one in a single step, the result is added to the carry from the previous step, as illustrated in Fig 7 unit’s place digit is the result bit and next digit bit is the carry to the next step. At step 1 carry is taken as 0.

Fig. 7: Multiplication of 2 Digit Decimal Numbers

2) Nikhilam Sutra:
This multiplication scheme is 64-bit wide and is programmed using Verilog HDL. Implementation of this method is done using FPGA Spartan-6 kit. Device utilization is improved by Nikhilam sutra as compared to the other two methods (Braun multiplier, Urdhva-Triyabhyam sutra). And also this method is effective in terms of Slices, LUT’s, and IOBs. Let’s consider one example, in order to multiply 8 by 9. The first step is to select nearest base for the multiplicand and multiplier, which is power of 10. The nearest base for these values (multiplicand and multiplier) is 10. Place the numbers (8 and 9) above and below on the left-hand side of a table as:

\[
\begin{array}{c}
8 \\
9 \\
\end{array}
\]

Subtract 8 (multiplicand), and 9 (multiplier) from the base and place the subtracted value on the right-hand side of the table as:

\[
\begin{array}{c}
8 - 2 \\
9 - 1 \\
\end{array}
\]

The next step is to generate the common difference, which is the final left hand side digit. This digit is obtained by subtracting one digit from the left column with the one digit of right column (crosswise) as shown below:

\[
\begin{array}{c}
8 \\
9 \\
\end{array}
\]

Common difference \(7/2\)

The final right hand side digit is the vertical multiplication of the base differences in right column (in this example 2 by 1).

\[
\begin{array}{c}
8 - 2 \\
9 - 1 \\
7/2 \\
\end{array}
\]

Thus, the final result is 72.

Fig. 8: Architecture of Nikhilam Sutra

C. Truncated Multiplier:
The truncated multiplier is an array multiplier. The architecture consists of AND gates and two types of fixed input multiplexers. One is with inputs 0, 0, 0, Ci, where Ci is carry bit input of multiplexer which will take care of carry generated in the previous column in architecture or from inputs by ANDing them and other with 0, xi, yi, si where xi,yi are the input bits of 8 bit numbers and si is the sum bit taking summation result coming from the previous column of multiplexer. The figure 8 shows The RTL view for Truncated multiplier.
IV. RESULTS

Simulations result of these multipliers are performed successfully on Xilinx 13.1 for Spartan 3 FPGA kit.

<table>
<thead>
<tr>
<th>Slices used</th>
<th>Braun Multiplier</th>
<th>Urdhva Triyakbhyanam Sutra</th>
<th>Nikhilam sutra</th>
<th>Truncated multiplier</th>
<th>PCT Truncated multiplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>11632 out of 63400</td>
<td>255 out of 63400</td>
<td>210 out of 63400</td>
<td>51 out of 63400</td>
<td>38 out of 63400</td>
<td></td>
</tr>
<tr>
<td>IOBs</td>
<td>256</td>
<td>387</td>
<td>340</td>
<td>128</td>
<td>107</td>
</tr>
<tr>
<td>Delay</td>
<td>54.672 ns</td>
<td>46.643 ns</td>
<td>44.43 ns</td>
<td>36.9 ns</td>
<td>32.1 ns</td>
</tr>
<tr>
<td>Area</td>
<td>513696 kilobytes</td>
<td>421856 kilobytes</td>
<td>401051 kilobytes</td>
<td>400108 kilobytes</td>
<td>368190 kilobytes</td>
</tr>
</tbody>
</table>

Table 1: Slices, Delay, IOBS Used For Various Multiplier

The above table shows that comparison result of standard (Braun, vedic multiplier) with the proposed multiplier (PCT truncated multiplier) and from the simulation result proved that delay in PCT truncated multiplier is less compared to standard and Truncated multiplier.

<table>
<thead>
<tr>
<th>Power (mw)</th>
<th>Truncated multiplier</th>
<th>PCT Truncated multiplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>360</td>
<td></td>
<td></td>
</tr>
<tr>
<td>346</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Power Consumption Truncated and PCT Multiplier

V. CONCLUSION

There are totally five multiplier implemented in this project (Braun multiplier, Urdhva Triyakbhyanam Sutra, Nikhilam sutra, Truncated, PCT truncated multiplier) these multipliers are tested and simulated successfully. By observing simulated result of each multiplier, a PCT truncated Multiplier is occupying more area but its power consumption is better than Truncated Multiplier with less propagation delay. A PCT truncated multiplier can be made most effective in terms of power, area, propagation delay than the proposed standard multipliers like braun multiplier, vedic multiplier, and truncated multiplier.

REFERENCES