Design and Implementation of High Speed and Low Power Multiplier using Urdhwa Tiryagbhyam Sutra

M. Padmaja¹,M.Tech
Department of ECE, K.G.R.C.E.T,
Moinabad, Hyderabad, AP

Anuradha Shankar ²,
Student, Department of ECE, K.G.R.C.E.T,
Moinabad, Hyderabad, AP

A. Saida³,M.Tech
Department of ECE, K.G.R.C.E.T, Moinabad,
Hyderabad, AP

Arjun Singh⁴,Student,
Department of ECE, K.G.R.C.E.T,
Moinabad, Hyderabad, AP

Abstract—A processing unit devotes considerable amount of its processing time in performing arithmetic operations and multiplication operation plays a vital role in this. As a multiplier unit is required in most real time processing applications, so higher throughput multiplication operations are important to achieve desired performance. An efficient multiplier design is proposed using vedic mathematics sutra :- Urdhwa Tiryagbhyam, 3:2 compressors[1][2] and a 4 bit novel adder that reduces delay and power.

KEYWORDS: Multiplication, Urdhwa Tiryagbhyam, 3:2 compressors, 4 bit novel adder.

I. INTRODUCTION

Multipliers are an integral part of most processing units hence the performance of processors greatly depend upon the functioning of their multiplication units.

Multiplication is the process of adding a number of partial products. Multiplication algorithms differ in terms of partial product generation and partial product addition to produce the final result. Higher throughput arithmetic operations are important to achieve the desired performance in many real time signal and image processing applications.

Several new architectures have been proposed for improving the functioning of multiplier units to meet the constraints of reducing the delay, power consumption, regularity of layout and hence less area or even combination of them in one, thus making them suitable for various high speed, low power and compact VLSI implementations. Though an efficient multiplier design is yet to come. In order to address the disadvantages associated with multiplier architectures, vedic mathematic approach was proposed. Multipliers were designed using Urdhwa Tiryagbhyam

[3]. In this paper, an even more efficient approach to improve multiplier units compared to the vedic multipliers is being proposed.

II. EXISTING TECHNIQUES

A. Array Multiplier

Array multiplier is well known due to its regular structure. It is an efficient layout of a combinational multiplier.

Multiplier circuit is based on add and shift algorithm. Each partial product is generated by the multiplication of the multiplicand with one multiplier bit. The partial product are shifted according to their bit orders and then added.

FIGURE 1: A 4 bit array multiplier

Consider an array multiplier for two binary numbers A and B, of n bits each as shown in figure 1 below. There are n² summands that are produced in parallel by a set of n² add gates. An \( \times n \) multiplier requires \( n(n-1) \) full adders and
n°and gates. Although the method is simple as it can be seen from this example, the addition is done serially as well as in parallel.

Array multipliers have high power consumption as well as number of components required. Delay is the time taken by the signals to propagate through the gates and in array multiplier, the worst case delay would be \((2n+1) t_d\) due to the gates that forms the multiplication array.

Thus array multipliers are less economical with more hardware complexity.

B. Vedic Multipliers: using Urdhwa Tiryagbyam sutra and 4 bit novel adder

Vedic multiplier designed using Urdhwa Tiryagbyam sutra and 4 bit novel adder [4] is as shown below in figure 2. The novel 4 bit adder performs the addition of 4 bits at a time and produces three output bits. These three output bits comprise of one sum bit and two carry bits.

The 4 bit adder adds the four input bits at a time and the speed of the multiplication increases.

It takes 3 inputs A, B, C to generate 2 outputs, the sum and the carry bits. Equations for sum and carry bits are governed by 1 and 2 as:

\[
\text{Sum} = (A \oplus B) \cdot \bar{C} + (A \oplus B) \cdot C \quad (1)
\]

\[
\text{Carry} = (A \oplus B) \cdot C + (A \oplus B) \cdot A \quad (2)
\]

This compressor is built using xor-xnor and multiplexer modules. We are replacing the full adders in the vedic multiplier design using novel 4 bit adder by 3:2 compressor. Even though a 3:2 compressor works same as full adder, the difference lies in propagation delay. A full adder needs 2 half adders which are in turn built using ‘xor’ and ‘and’ gates. The delay produced by a full adder is 0.027 ns whereas a 3:2 compressor adder produces a delay of only 0.019 ns

By using this 4x4 multiplier we can design the architecture for 8x8 multiplier also. Though the multiplier reduces the design complexity and power drastically the delay can still be reduced further.

III. Proposed multiplier

In this proposed multiplier design, we are introducing compressors in the existing vedic multiplier using novel adder. A compressor is a device that reduces the combination of input bits at the output. Shown below in figure 3, is a 3:2 adder compressor that functions similar to a full adder.

Hence we are improving the delay and power consumption very efficiently in the above.
multiplier architecture though the area constraint can still be improved further using other techniques

IV. Results

A. Simulation Results

Using the Xilinx 12.2 version and Spartan 3 FPGA kit, the simulation results were found as shown below

(i) For 4 bit proposed multiplier

FIGURE 6: Simulation output for 5x12,3x5, 12x2, 15x15

(ii) For 8 bit proposed multiplier

FIGURE 7: Simulation output for 204 x 204, 204 x 239, 204 x 236, 239 x 255

B. Delay Tables and Graphs

(i) For 4 bit proposed multiplier

<table>
<thead>
<tr>
<th>Technology in nm</th>
<th>Array multiplier</th>
<th>Vedic multiplier</th>
<th>Proposed Vedic multiplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>65</td>
<td>0.160 ns</td>
<td>0.103 ns</td>
<td>0.092 ns</td>
</tr>
<tr>
<td>90</td>
<td>0.339 ns</td>
<td>0.219 ns</td>
<td>0.199 ns</td>
</tr>
</tbody>
</table>

FIGURE 8: Delay table, graph for 4 bit

(ii) For 8 bit proposed multiplier

<table>
<thead>
<tr>
<th>Technology in nm</th>
<th>Array multiplier</th>
<th>Vedic multiplier</th>
<th>Proposed Vedic multiplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>65</td>
<td>0.264 ns</td>
<td>0.166 ns</td>
<td>0.149 ns</td>
</tr>
<tr>
<td>90</td>
<td>0.569 ns</td>
<td>0.370 ns</td>
<td>0.340 ns</td>
</tr>
</tbody>
</table>

FIGURE 8: Delay table, graph for 8 bit
C. RTL and Technology Schematics

(i) For 4 bit proposed multiplier

(ii) For 8 bit proposed multiplier

FIGURE 9: Schematics for 4 bit

FIGURE 10: Schematics for 8 bit
D. Power comparison table and graph

<table>
<thead>
<tr>
<th>Multiplier</th>
<th>Power in (mw)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Array</td>
<td>0.580</td>
</tr>
<tr>
<td>Vedic</td>
<td>0.484</td>
</tr>
<tr>
<td>Proposed Vedic</td>
<td>0.418</td>
</tr>
</tbody>
</table>

FIGURE 10: Power comparison in different multipliers

V. CONCLUSION

The proposed vedic multiplier using 4 bit novel adder and 3:2 compressor has produced an improved performance compared to its predecessors by reducing the delay and power consumption. These multipliers can improve the performances of applications in which they are used.

VI. FUTURE SCOPE

Though hardware area has reduced by only small percent, it can be further improved. The same can be implemented for higher bits also.

VII. REFERENCES

1. Sushma R. Huddar and SudhirRao