Design Approach of High Performance Arithmetic Logic Unit using Pipelined Multiplier based on Vedic Mathematics

S. R. Parimal 1, Prof. M.N. Thakare 2, Prof. S. R. Vaidya 3

1Research Scholar 2, 3Assistant Professor

1, 2Department of Electronics & Communication Engineering

1, 2SDCE, Selukate, Wardha 3BDCOE, Sewagram, Wardha

Abstract—All of us know that ALU is a module which can perform arithmetic and logic operations. The reason behind choosing this topic as a research work is that, ALU is the key element of digital processors like as microprocessors, microcontrollers, central processing unit etc. Every digital domain based technology depends upon the operations performed by ALU either partially or whole. That’s why it highly required designing high speed ALU, which can enhance the efficiency of those modules which lies upon the operations performed by ALU. The speed of arithmetic is of extreme importance and depends greatly on the speed of multiplier. Therefore the technologies are always looking for new algorithm and hardware so as to implement this operation in much optimized way in the terms of area and speed. Vedic Mathematics deals with various branches of mathematics like arithmetic, algebra, geometry etc in computation algorithm of the coprocessor which will reduce the complexity of execution time, area and power consumption etc. Multiplier based on Vedic Mathematics is one of the fast and low power multiplier. In this paper arithmetic logic unit is design with the pipelining technique based on vedic mathematics to improve the execution speed and consuming power.

Key words: Arithmetic Logic Unit, Vedic Mathematics, Pipelining

I. INTRODUCTION

Pipelined Multiplier is designed mainly to optimize speed of the multiplier which is the major requirement in many applications. Pipelining will enhance the performance of the multiplier. The multiplier is implemented for the low contrast image enhancement. There are so many multiplication algorithms exist now a days at algorithmic and structural level. Work proved that Vedic multiplication technique is the best algorithm in terms of speed. Further we have seen that the conventional Vedic multiplication hardware have some limitations. So to overcome those limitations to design the Vedic multiplier with the use of unique addition tree structure, which is used to add partially generated products [1]. Ancient Indian system of mathematics known as Vedic mathematics can be applied to various branches of engineering to have a deeper insight into the working of various formulae. A typical processor devotes a considerable amount of processing time in performing arithmetic operations, particularly multiplication operation. The design of a low power high speed algorithm for arithmetic logic unit using this ancient mathematics technique also their hardware implementation. Employing this technique in the computation algorithm of the co-processor has reduced the complexity, execution time, area and power. The hardware can be implemented using Verilog HDL[2]. Presenting a multiplier, in which the basic multiplication is perform using one of the technique of Vedic Mathematics and the accumulation of partial products is done using specific design. In this technique intermediate product are generated in parallel that makes multiplication faster. Basic multiplier architecture is based on Vedic technique and accumulation is done using carry save adder, which gave better performance on comparison with other multiplier, that found the design works with much less delay. The synthesized design using Xilinx ISE tool [3].

The multipliers are the basic and essential building blocks of many high performance systems. Multiplication is frequently used operation which is currently implemented in many processors. Pipelined has long been kwon as efficient technique for optimizing the computational time. In this conclude that it gives comparison of clock frequency between pipeline and non-pipelined multiplier [4]. Improvement of speed over the conventional designs and then the result achieved by the proposed method has been compared by Karatsuba Multiplier. For the implementation of arithmetic unit using optimized Vedic multiplier which is not only useful for the optimizing the arithmetic unit of ALU but also is useful in the field of digital signal processing directly[5]. The objectives of good ALU to provide a physically compact good speed & low power consumption unit. Being part of arithmetic processing unit, multipliers are in extremely high demand on its speed & low power consumption. To save significant power consumption of ALU design it is good direction of operations there by reducing a dynamic power which is a major part of total power dissipation. The proposed arithmetic unit is coded in verilog HDL synthesized and simulated using Xilinx software. These experiment generates the result that pipelined ALU is more efficient than the simple ALU[6].

The block of pipelined multiplier speed enhancement leads to the whole system speed enhancement. Pipelining is a technique called overlapping of multiple instructions during execution .By using this technique the optimized speed and low power system is being achieved. Also by using these technique the delay is to be reduced on comparing with the non pipelined structure [7]. High speed Vedic multiplier architecture which is quite different from the Conventional method of multiplication like addition and shift. Further, the Verilog HDL coding of Urdhva tiryakhyam Sutra. This presents the design of high speed Vedic Multiplier using the techniques of Vedic Mathematics that have been modified to improve performance. A high speed processor depends greatly on the multiplier as it is one of the key hardware blocks in most digital signal processing systems as well as in general processors[8]. A highly efficient method of multiplication – “Urdhva Tiryakhyam Sutra” based on Vedic mathematics. It is a method for hierarchical multiplier design which clearly presents the computational advantages offered by Vedic methods. It is observed that the Vedic multiplier is much more efficient than Array and Booth multiplier in terms of execution time[9].
II. VEDIC MATHEMATICS

Vedic mathematics is part of four Vedas (books of wisdom). It is part of Sthapatyaa- Veda (book on civil engineering and architecture), which is an upa-veda (supplement) of Atharva Veda. It covers explanation of several modern mathematical terms including arithmetic, geometry (plane, co-ordinate), trigonometry, quadratic equations, factorization and even calculus. His Holiness Jagadguru Shankaracharya Bharati Krishna Teerthaji Maharaja (1884-1960) comprised all this work together and gave its mathematical explanation while discussing it for various applications. Swamhiji constructed 16 sutras (formulae) and 16 Upa sutras (sub formulae) after extensive research in Atharva Veda. The very word „Veda“ has the derivational meaning i.e. the fountainhead and immutable storehouse of all knowledge. Vedic mathematics is the name given to the ancient system of mathematics or, to be precise a unique technique of calculations based on simple rules and principles with which many mathematical problems can be solved, be it arithmetic, algebra, geometry or trigonometry. The system is based on 16 Vedic sutras or aphorisms, which are actually word formulae describing natural ways of solving a whole range of mathematical problems. The beauty of Vedic mathematics lies in the fact that it reduces the otherwise cumbersome calculations in conventional mathematics to a very simple one. This is so because the Vedic formulae are claimed to be based on the natural principles on which the human mind works. This is a very interesting field and presents some mathematical explanation while discussing it for various applications.

Sub sutras or Corollaries

III. ERROR CORRECTION CODE

The multiplier is based on an algorithm Urdhva Tiryakbhyam(Vertical & Crosswise) of ancient Indian Vedic Mathematics. Urdhva Tiryakbhyam Sutra is a general multiplication formula applicable to all cases of multiplication. It literally means “Vertically and crosswise”. It is based on a novel concept through which the generation of all partial products can be done with the concurrent addition of these partial products. The parallelism in generation of partial products and their summation is obtained using Urdhva Tiryakbhyam.

Multiplication of two decimal numbers- 325*738
To illustrate this multiplication scheme, let us consider the multiplication of two decimal numbers (325 * 738). Line diagram for the multiplication is shown in Figure 2. The digits on the both sides of the line are multiplied and added with the carry from the previous step. This generates one of the bits of the result and a carry. This carry is added in the next step and hence the process goes on.

![Fig. 1: Multiplication of two decimal numbers by Urdhva Tiryakbhyam Sutra](image-url)
IV. PIPELINING

Pipelining is one of the popular methods to realize high performance computing platform. Pipelining is a technique where multiple instruction executions are overlapped. It comes from the idea of a water pipe: continue sending water without waiting the water in the pipe to be out. By pipelining the unit of a system we can produce result in every clock cycle. It leads to a reduction in the critical path. It can either increase the clock speed (or sampling speed) or reduces the power consumption at same speed in a DSP system. Pipelining is a concept to reduce the delay in the critical path. By eliminating the delay in the critical path the speed and throughput is increased [9]. The Pipeline pattern uses parallel tasks and concurrent queues to process a sequence of input values. Each task implements a stage of the pipeline, and the queues act as buffers that allow the stages of the pipeline to execute concurrently, even though the values are processed in order. You can think of software pipelines analogous to assembly lines in a factory, where each item in the assembly line is constructed in stages. The partially assembled item is passed from one assembly stage to another.

V. IMPLEMENTED WORK

Using the 2-bit Vedic Multiplier we generate 4-bit Vedic Multiplier, 8-bit Vedic Multiplier, 16-bit Vedic Multiplier, 32-bit Vedic Multiplier that all multipliers shows the result in terms of power dissipation and no. of delay in table. Using that Vedic Multipliers we generated a 32×32 Bit ALU.

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Parameter</th>
<th>Delay in Ref</th>
<th>Proposed Pipelined ALU</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Execution time</td>
<td>0.885ns</td>
<td>0.83ns</td>
</tr>
<tr>
<td>2.</td>
<td>Speed</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>3.</td>
<td>Delay</td>
<td>2.370ns</td>
<td>7.308ns</td>
</tr>
<tr>
<td>4.</td>
<td>Power</td>
<td>0.020W</td>
<td>0.042W</td>
</tr>
</tbody>
</table>

Table 1 shows the result of 32-bit Non Pipelined ALU & Pipelined ALU in terms of Power Dissipation, Delay, Speed & Execution Time.

![Fig. 2: Schematic of 32×32 bit Pipeline ALU in Xilinx ISE tool](image)

Firstly for ALU we can see that result in terms of 4 bit Vedic Multiplier, 8 bit Vedic Multiplier, 32 bit Vedic Multiplier. It gives good response for the power consumption and delay, speed. Also we compared these result with 32×32 bit Vedic Pipeline ALU & Non Pipelined ALU in terms of the power dissipation and delay.

The Arithmetic Logic Unit with and without pipelining multiplier that designed and verified using Xilinx ISE tool. For both ALU we can see that result in terms of power, delay, speed & area. Also we compared result with 32-bit Non Pipelined ALU & Pipelined ALU in terms of power, delay, speed & area.

VI. CONCLUSION

We have designed Pipelined ALU based on Ancient Indian Vedic Mathematics for less delay, low power and high performance ALU. We use Vedic multipliers generated 32×32-bit ALU according to area, delay and power dissipation. Also all the multipliers are generated using a Ancient Indian Vedic Mathematics technique and we will see the result of multipliers. But when we compared the result of Pipelined and Non Pipelined ALU with 32×32-bit then we see that speed is increasing because the buffer is added in to it because of the pipelining concept. When we see the result in terms of delay for proposed work 2.370ns for Pipelined ALU. But in reference work the result in terms of delay 28.27ns. So the delay can reduced by using the pipelining. By comparing previous work and proposed work The timing delay is greatly reduced to 25.9 % for Pipelined ALU based on Vedic Mathematics as compared to other design. The low power dissipation and delay of proposed Pipelined ALU makes it more useful in less power requirement and high speed applications.

REFERENCES


