Design and Synthesis of Systolic Array Architecture for Matrix Multiplication

Shreedevi M. Ganji¹, Shrikanth K. Shirakol²
¹IVth Semester, M.Tech, Department of Electronics and Communication, SDMCET, Dharwad, INDIA
²Assistant Professor, Department of Electronics and Communication, SDMCET, Dharwad, INDIA

ABSTRACT

In most of the signal and image processing applications, matrix multiplication plays very important role. It is the very basic operation used in so many DSP applications. Through this paper an effective design for the Matrix Multiplication is demonstrated using Systolic Architecture. Here the computing speed is increased by the use of pipelining and parallel processing together as a single concept. Here, the code is written for matrix multiplication without systolic architecture and matrix multiplication with systolic architecture in Verilog HDL, compiled and simulated by using Xilinx ISE 14.2i and targeted to the device xc3s500e-5-ft256 then finally the designs are compared to each other to evaluate the performance of proposed architecture. In the proposed Matrix Multiplication with systolic architecture vedic multiplier is used to speed up the computation speed.

Keywords— systolic array, vedic multiplier, processing element(PE).

I. INTRODUCTION

Matrix multiplication is the very basic operation in DSP and image processing applications. Basically implementation of multiplication in hardware as well as in software is tough task, in that to the operations like matrix multiplication, FFT, DFT, DCT calculations are further more complex problems. Which include addition together with the multiplication so it is the basic need to find the efficient ways for faster calculations of multiplication as well as method to handle the carry generation and propagation in adders one of the way to speed up the operations is systolic array architecture[1].

In computer architecture a systolic architecture is a array of processing elements it forms a pipelined network arrangement of processing elements called as cell. systolic array is a technique of computing parallel, it takes incoming inputs and compute the results and stores them separately. In the systolic array processing elements acts like central processing units. After calculating the results or partial product, the information is shared between the neighbors very immediately. A systolic array may be triangle, hexagonal or rectangle, but most of the time is rectangle. In each processing element there will be one adder, multiplier and accumulator, Here the adder and the multiplier must be designed very efficiently. So in order to maximize the speed and to reduce the area vedic multiplier and modified carry save adder is adapted in processing elements which will includes the pipelining and parallel processing as a single concept.

II. METHODOLOGY

There are three types of multiplier architectures serial multiplier, parallel multiplier and serial parallel multiplier. serial multiplier is comparatively less complex to implement and it consumes minimum area for hardware implementation but it takes too much of time to compute the product because of handling the carry propagation, the delay is due to the carry propagation. In parallel multiplier[2] this problem has been eliminated by using special kinds of carry propagation circuits nothing but special kinds of adders which includes the techniques for maintaining the carry propagation in an intelligent manner. Parallel multipliers take the structure of array or tree so they are also called as array or tree multipliers. These parallel multipliers are high speed but requires larger chip area. And the third one is serial-parallel multiplier is trade off between above two which have the features of both multipliers, time consuming like serial multiplier and area consuming like parallel multipliers. The proposed systolic array architecture is the kind of parallel matrix multiplication.
The Systolic Array is characterized by processing the data input in pipeline and it comprises of Processing elements which are regular simple and arranged in arrays. It uses the parallel matrix multiplication[3], and neighbor PEs are connected to each other by shortest line. So there is no necessary of storing the mass data before processing the inputs. Internal communication delay is reduced greatly by decreasing the distance between neighbor processing elements in an array. And also the utility of processing elements is improved. It also removes time consumption for controlling the establishment of data stream. In, this research, the PE is replaced with Multiplication and Accumulation (MAC) to enhance the speed and reduce the complexity of Systolic Architecture.

A basic principle of systolic array architecture is shown in figure1. It tells that in normal operations the processing element performs one operation per unit time but in systolic array architecture the array of processing element performs the operation parallely and in pipelined fashion. That is it performs multiplication in one second.

A. Conventional design principle
B. Systolic array design principle

Figure1. Basic principle of systolic array architecture

III. IMPLEMENTATION SCHEME

The systolic array is already used for the matrix multiplication. In previous papers the processing element is represented by multiplier and accumulator element, and each processing element performs the basic multiplication, shift and addition operations but here in this paper the processing element is represented by vedic multiplier and accumulator or register to store the partial products. Here the 4 bit vedic multiplier is designed and it is used four times to form the 8 bit vedic multiplier to compute the matrix multiplication of inputs 8 bit each. And also the matrix multiplication is done conventionally and at last the results of conventional matrix multiplication method, systolic method with basic multiplier, and systolic method with vedic multiplier is compared.

3.1 Systolic array architecture

A systolic array architecture is a array of processing elements, it comprises of processing elements connected to nearest neighbors at top, right, left and bottom and forms a mesh like topology. Each cell performs a sequence of simple mathematical operations on the data values that passes between them. At each step PE takes incoming data from one or more neighbors may be from left and top, processes it and in the next step result is given or outputted in the opposite direction that is right and bottom[4]. The Proposed two dimensional systolic Architecture for 3 by 3 matrixes is given in Fig 2.

3.2 Vedic multiplier

Multiplication is very important operation in almost all DSP applications. The speed is the most important thing we have to consider in design of each and every component. So here the delay can be reduced to some extent by using vedic multiplier based on vedic mathematics.

Vedic multiplier based on vedic multiplier is one of the low power and fast multiplier. The Vedic mathematics is part of four Vedas (books of wisdom). It is part of Sthapatya- Veda (book on civil engineering and architecture), which is an upa-veda (supplement) of Atharva Veda. It gives explanation of several mathematical terms including geometry, arithmetic, quadratic equations, trigonometry, factorization and even calculus. Jagadguru Shankaracharya Bharati Krishna Teerthaji Maharaja studied the vedic mathematics and he had given the mathematical formulae and its explanation for various applications. He defined 16 sutras (formulae) and 16 upa sutras or sub formulae after extensive research in Vedas[5].

Figure2. The proposed two dimensional systolic architecture for 3*3 matrix multiplication.

There are sixteen sutras in vedic mathematics that can be applied directly to the mathematical operations. In this, Urdhva tiryakbhyam Sutra is almost same as that of popular array multiplier. This sutra is very effective that it reduces the MXM multiplier structure into an efficient
4X4 multiplier structures. And these smaller structures can be used to build further higher bit multipliers. In this paper urdhva tiryakbhyam sutra is used which is a common multiplication formula that can be applied to all cases of multiplication. The meaning of this sutra is “Vertically and Crosswise”. In conventional or general (shift and add) 4X4 multiplication method to get the final product four partial results are to be added, but it is not so in this sutra it can be done in a single line[6]. Thus it shows that the number of steps required to calculate the final product is reduced and so there is a reduction in the computation time and the speed of the multiplier is increased. Steps for 4 bit vedic multiplication using the above mentioned sutra is shown in the figure3. The steps shows the parallel and crosswise multiplication[7] which is as explained above.

The 8 bit vedic multiplier design is shown in the figure4. Here The 8-bit incoming input sequence is divided into two 4-bit numbers and given as inputs to the 4-bit multiplier blocks (a[3:0] & b[3:0], b[3:0] & a[7:4], b[7:4] & a[3:0], a[7:4] & b[7:4]). Four similar 4 bit vedic multipliers are used to calculate the partial products and those partial products are added using the adders. And here three adders are used to calculate the final product which are named as adder1,adder2 and adder3. And the same is represented in the figure4. From each multipliers the few bits of products are taken. i.e. the four LSB product bitsP[3:0] are taken directly from first multiplier. The results of second and third multipliers are added using adder1, the first four bits of the result of adder1 are given to the adder3 and next four bits are given to the adder2, and the next four bits of the final product are taken from the result of adder2, now totally it forms 8 bits of final product. Now whole partial result of last multiplier is given to the adder3, as shown in the in the figure4. The next higher eight bits of the final product are taken from the output of the third adder. Now combining all the product bits it forms sixteen bits of final product.

IV. RESULTS AND DISCUSSION

The matrix multiplication is implemented in both the ways i.e. conventional and systolic architecture. As described above the code is written in verilog HDL, logic and simulation is verified using Xilinx 14.2. the simulation results says that, implementation of matrix multiplication
requires less number of clock cycles than the conventional method. This section gives the simulation results, timing report, top module and RTL schematic of both conventional and proposed systolic array architecture for matrix multiplication. Number of device utilization is less for the proposed architecture as compared with the conventional method, so the area can be reduced in the hardware design of matrix multiplication for proposed method. By seeing the timing report the propagation delay for proposed architecture is found to be less. So the speed can be enhanced and area can be efficiently reduced by using pipelining and parallel processing as a single concept in systolic architecture.

<table>
<thead>
<tr>
<th>Logic utilization</th>
<th>Used</th>
<th>Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional method</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Proposed method</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 2. Performance analysis of systolic array architecture method for matrix multiplication.

<table>
<thead>
<tr>
<th>Name of the component</th>
<th>Conventional method</th>
<th>Systolic architecture method with normal multiplier</th>
<th>Systolic architecture with vedic multiplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delay</td>
<td>9.831ns</td>
<td>4.757ns</td>
<td>2.819ns</td>
</tr>
</tbody>
</table>

### V. CONCLUSION

Through this paper the systolic array is designed for the matrix multiplication. The vedic multiplier used in each processing element reduces the steps of multiplication operation, and also the parallel processing and pipeling concepts makes the operations faster. The complexity also reduced and less number of components are used as compared with the conventional matrix multiplication. Finally the delay is also reduced as compared with conventional and also systolic method with normal multiplier.

### REFERENCES