Implementation of Low power systolic based RLS adaptive Filter using FPGA

Purushothaman A
DA-IICT, Gandhinagar, India
Email:200721004@daiict.ac.in

Dr.C.Vijaykumar
DA-IICT, Gandhinagar, India
Email: vijaykumar.chakka@daiict.ac.in
Telephone: (079)-30520000-590

Abstract—We implemented RLS based Adaptive filter in FPGA by employing systolic architecture with clock gating. For obtaining high performance computation on a large array of cells, we are using systolic array. Recursive Least Square (RLS) is one of the algorithms that can be used to update array weights in adaptive array antennas. Although the calculation load is large, it achieves fast convergence and is thus effective under fading environments; it shows great promise in mobile communication applications. We first constructed a fully functional testbed in which the RLS algorithm is implemented using FPGA. Then, we proposed a new architecture of systolic arrays that allows a significant reduction of number of processing clock cycles required for updating the array weights. We demonstrated that it is possible to reduce the circuit scale dramatically without impairing the processing speed by reusing circuits, making use of the symmetric property of internal cells constituting the systolic arrays. In this paper we propose a scalable systolic array architecture which consists of another systolic array, produces high performance, and can be directly adopted in the VLSI design including regular and local interconnection and functional primitives that are typical for a systolic architecture. The cell of a systolic array derived through projection and scheduling upon the dependence graph, from the given behavior can be designed as another systolic array, and this systolization procedure of implementing cell as another systolic array could be apply repetitively in a hierarchical manner until a cell having only primitive operators is obtained. Today the power consumption of integrated circuits (ICs) is considered as one of the most important problems for high performance chips. Many low power techniques have been proposed during the last 15 years [1]. The micro-power systolic array model is described writing efficient register transfer logic (RTL) code from a low- power standpoint using clock gating technique.

I. INTRODUCTION

In the area of special purpose architecture for digital signal processing, systolic arrays are recognized as a standard for high performance. A systolic array formed by interconnecting a set of identical data-processing cells in a uniform manner is a combination of an algorithm and a circuit that implements it, and is closely related conceptually to arithmetic pipeline. In a systolic array, data words flow from external memory in a rhythmic fashion, passing through many cells before the results emerge from the array’s boundary cell and return to external memory. The external memory connected to the systolic array’s boundary cell stores both input data and results. Upon receiving data words, each cell performs same operation and transmits the intermediate results and data words to adjacent cells synchronously. The underlying principle of systolic array is to achieve massive parallelism with a minimum communication overhead, and generally speaking, a systolic array is easy to implement because of its regularity and easy to reconfigure because of its modularity. With advances in VLSI and FPGA technologies, the systolic arrays have progressed from fixed function to general purpose architecture and data and partial results flow from 1 or 2 directions to multiple directions. In a systolic array which implements the recursive equations over a large array of cells, it is needed to organize the cell of a systolic array itself as another systolic array, that is, super-systolic array to raise the cell utility. This paper proposes a bit-level systolic RLS adaptive filter with an FPGA-based bit-serial semi-systolic multiplier which twists on the shift and add multiplier by positioning upper and lower half of the serial multiplier side by side physically in floor planning, instead of linearly. The proposed design is better suited to FPGA implementation than bit-level systolic RLS adaptive filter with a bit-serial systolic multiplier in each cell in terms of hardware complexity, PondR and performance. In order to achieve high performance VLSI computing systems three primary goals have to be successfully solved. The first one relates to highspeed computational systolic array capability, the second one is with an efficient data transfer among computational building blocks, and the third one is concerned with micro-power consumption. Processor arrays, as a regular arrangement of systolic arrays, are good candidate accelerator-architectures that are used in many VLSI computing systems with aim to achieve high computational and communication performance. A challenging problem which should be taken into consideration by the VLSI designers now relates both to increase performance and improve energy efficiency. The architectural systolic array design for low power is in focus of our interest in this paper. In particular we will consider RTL synthesis of a simple micro-power systolic array which represents a constituent of a complex VLSI computational system.

II. IMPLEMENTATION OF RLS ALGORITHM IN FPGA

The principle of adaptive array antennas is to weigh and combine signals of multiple antenna elements for the purpose of aiming the beams of an antenna in the directions of desired waves and forming nulls in the directions of interferences.
An adaptive array antenna adopting a Minimum Mean Square Error (MMSE) algorithm [1] allows eliminating delayed waves without requiring information of the arrival direction of the desired waves; it is thus possible to achieve high-quality communication in multipath environments as well. Least Mean Square (LMS) and RLS algorithms are the most commonly used for weight optimization, and the RLS algorithm is typically used in fading environments that change over time due to its fast convergence. Table 1 shows the specifications of the testbed; 69\% (approximately 410,000 gates) of total area was required for the implementation in the FPGA board. A significant improvement can be observed compared to the case where only one antenna is used. Note that the difference between the computer simulation (theoretical values) and experimental data is caused by factors other than the signal processing (antenna elements, measurement jig, cables and so on). As the results of evaluations by using the testbed, we confirmed that the number of updates required before weight convergence was in the order of 10 to several tens, and the processing required for one weight update was approximately 400 clock cycles.

III. IMPLEMENTATION OF SYSTOLIC ARRAYS

Systolic arrays represent a calculation architecture in which calculations are performed in parallel by arranging circuits (cells) that perform individual, simple calculations in a regular pattern and pouring in the data required for the calculation in a pipelined manner. This parallel approach can improve the processing speed to a large extent and the uniform structure provides excellent expandability. Moreover, since the majority of the cells are only connected directly to adjacent cells and the data exchange is local, they are highly suited for implementation using Large Scale Integration circuits (LSI). Figure 1 shows the architecture used for implementing the RLS algorithm using systolic arrays [3]. The figure shows a case with four array elements. $x_1(i)$, $x_2(i)$, $x_3(i)$ and $x_4(i)$ represent the $i$th sample point of the signals of each element and $y(i)$ represents the $i$th sample point of the reference signal used for identifying the desired signal. The systolic arrays are mainly configured using two types of cells, boundary cells and internal cells. All cells operate under the control of a single clock, and the data propagates through the cell arrays in synchronization with this clock, so that the processing is performed in a manner similar to the way blood circulates in a human body in synchronization with the contraction of the heart. Indeed, the word systolic is an adjective meaning heart contraction. Each data point is a complex number, and two multiplications of complex numbers are included in the internal cell processing. Figure 1 shows a specific circuit configuration of internal cells composed from adders, subtracters and multipliers. fpadd, fpsub and fpmul indicate adders, subtracters and multipliers, respectively. Complex number multiplication is resolved using real number calculation by splitting each complex number into its real and imaginary parts. If the number of antenna elements is $M$, the number of internal cells, is $M \cdot (M+1)/2$ and thus increases with the square of the number of elements. M. Systolic arrays speed up the processing due to the parallel calculation, but have the issue that the circuit scale becomes extremely large if the number of elements is large. Multipliers and dividers consume the greater part of an area of digital circuits (circuit scale) and the possibility of making the circuit scale small thus depends on how the number of multipliers and dividers can be reduced. Note that, in general, multipliers and dividers are constructed using a number of adders proportional to the square of the number of digits of data handled (bit length), and they thus become very large compared to addition and subtracters. Moreover, multipliers and dividers also require significant processing time (delay) in digital circuits. Subtracters can be replaced with adders by reversing the sign of the input data in advance. We thus used a selector circuit to branch between the processing of the first and latter halves, thereby reducing the circuit scale of the internal cells to approximately half of the original size. Since the processing of the latter half of the internal cells uses the processing result of the first half as input signal, it is possible to reduce the circuit scale into approximately half, almost entirely without impairing the processing speed.

IV. CLOCK GATING

The clock gating technique is extensively used in the design of low-power circuit [2]. It involves dynamically shutting off the clock to portions of a design that are idle or are not performing useful computation. Fig. 2 depicts the concept of the clock gating using an AND gate. The basic idea is to AND the clock with an enable signal, so that the register receives a clock signal only when the enable is high. In this paper we
will consider an implementation of clock gating technique on simple systolic array as a constituent of 1D or 2D processor array. We define the processor array as an arrangement of systolic elements in an array (often rectangular) where data flows synchronously across the array between neighbors, usually with different data flowing in different directions. More details concerning systolic array can be found in [3].

V. IMPLEMENTATION EVALUATION

<table>
<thead>
<tr>
<th>No of Input Bits</th>
<th>Systolic Array</th>
<th>Micro power systolic array</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Gate count</td>
<td>Power(mW)</td>
</tr>
<tr>
<td>2</td>
<td>440,000</td>
<td>996.75</td>
</tr>
<tr>
<td>4</td>
<td>1,130,000</td>
<td>3932.89</td>
</tr>
<tr>
<td>8</td>
<td>1,960,000</td>
<td>7842.34</td>
</tr>
<tr>
<td>16</td>
<td>2,460,000</td>
<td>9334.34</td>
</tr>
</tbody>
</table>

Fig. 3 shows the result of implementing the RLS algorithm in FPGA using systolic arrays adopting the reusable configuration and clock gating proposed above for the internal cells. For the cases of two and four antenna elements, the required number of sampling data processing clock cycles was 20, which indicates that the weights are updated using a very small number of clock cycles. The total numbers of clock cycles required before convergence for the two and four antenna element cases were 320 and 540, respectively. Figure 3 shows a comparison between the beam patterns after convergence. As can be seen from the figure, a highly precise null was formed in the interference wave direction. The circuit scales were 440,000 gates and 1,130,000 gates, respectively. Conventionally, 19 Application-Specific Integrated Circuits (ASIC), each with 900,000 gates, were required in order to implement the RLS algorithm using systolic arrays for an antenna with 10 elements. With the proposed configuration, there is a strong chance of achieving the same performance using only 13 FPGAs (900,000 gates). It is evident that clock gating reduces around 29 percentage of power.

VI. CONCLUSION

This article described the implementation and evaluation result of the RLS algorithm using FPGA technology with clock gating. The newly proposed systolic array architecture can reduce the circuit scale into half without impairing the processing speed and clock gating can reduce the power considerably. In the future, we intend to investigate antenna elements and array configurations, as well as examine application of Multiple Input Multiple Output (MIMO) technologies, for the purpose of expanding the transmission capacity.

REFERENCES