Performance Analysis of FIR Filter Using Distributed Arithmetic

Shilpa saini, Sukhjit singh

Student, Department of Electronics and Communication Engineering
Golden College of Engineering and Technology, Gurdaspur, India
Assistant professor, Department of Electronics and Communication Engineering
Golden College of Engineering and Technology, Gurdaspur, India

Abstract

In this paper, various approaches to design FIR filter are discussed which are based on distributive arithmetic schemes. In terms of area utilization and operating frequencies, the unsigned DA scheme is used for the unsigned weight values. The signed DA scheme is employed for signed coefficients which analyze sign control accumulator instead of using ROM along with additional input, that utilizes a large memory. In this paper, a residue number system (RNS) based distributed arithmetic (DA) scheme is proposed for the calculation of inner products in the FIR filters. In this scheme the statistics or data content of distributed arithmetic LUT can be transformed into the RNS form. The suggested DA-RNS based FIR filter implementation established on binary code format (BCR) delivers effective calculation of modular inner products when compare to the DA approach which is based on simple scaling. The critical path considers the carry propagation for operating speed measurement. But in the case of RNS, the process of carry propagation is absent which provides high processing speed. Hence, increased speed between arithmetic blocks and encoding of large words into small words due to RNS provides small critical path.

Indexing terms

Distributed Arithmetic (DA), Digital Signal Processing (DSP), Residue Number System (RNS), Binary Code Residue (BCR), Field Programmable Gate Array (FPGA), and Look Up Table (LUT).

Academic Discipline and Sub-Disciplines

Department of Electronics and Communication Engineering, Very large scale integration.

SUBJECT CLASSIFICATION

Very large scale integration

APPROACH

In this paper, the optimization of digital FIR filter has been achieved in the direction of operating speed by considering the trade off factor between area and speed. In this paper work the parallel DA technique is proposed to increase the speed of FIR filter at slightly increase the area. The proposed scheme using parallel approach enhanced the operating speed as compare to signed and unsigned DA approach at the cost of slightly increase in the memory requirements.

INTRODUCTION

The employment as well as the accuracy of calculation of higher order filters becomes challenging task because of the complexity of the filters for real time realization. The systolic design representing the efficient hardware architecture supports the regularity, simplicity, and modularity of structure for intensive computations of DSP applications. They also posses a significant potential through high level concurrency which is achieved using pipelining or parallel processing, that helps in achieving high throughput rate. The possible maximum number of logic elements is limited due to the reason that multipliers acquire large area. The structures based on memory have various advantages like high throughput, abridged latency, and more steady as compared to multiple accumulate unit structures. The dynamic power consumption is less for memory read operations due to the reduced switching events as compared to the conventionally used multipliers. All the elements like delay units, multipliers, and adders are removed in order to utilize the memory-based structures through minimizing the area and reducing system latency. In various memory-based structures have been demonstrated for Digital Signal Processing applications and digital filters and Digital Signal Processing. The Residue number system (RNS) is used to enhance the speed of the arithmetic operations by removing the carry propagation step. It provides high operating frequency in DSP operation by reducing word length. In this paper, a Distributed Arithmetic (DA) based RNS FIR filter implementation is proposed to achieve the high throughput, minimum area utilization and high operating frequency. The computational workload to calculate the inner products of two vectors, adders and multipliers are basically used in the conventional FIR filters. But, the case is different for DA, as the adders and multipliers are replaced by the look up table (LUT), scaling accumulator, and shift registers, which provide computing structures for high throughput processing capability, area-time efficiency, and cost efficiency. The DA computation model is employed because it can compress and reduce the computational complexity of a tap filter from multiplications and additions into a LUT which uses $K \cdot \text{additions}$ additions to provide result in $K \cdot \text{bit time duration}$. A significant number of additions is removed in DA during filtering operations especially for high bit precision filters. The computational workload is...
further reduced, as the pre-computed partial sums of the filter coefficients are stored in form of residue number system in the memory table.

**DA Background**

The DA technique is a bit serial technique which is used to evaluate the inner products. It is basically attained by loading the pre-calculated aggregate of products in the LUT. The simple multiply and accumulate approach is used, which requires \( K \) multiple and accumulate (MAC) execution cycles to evaluate the inner product corresponding to the number of fixed coefficients (1). The inner product is expressed as:

\[
y = \sum_{k=0}^{K-1} C_k x_k \quad (1)
\]

where \( C_k \) denotes the fixed coefficient values and \( x_k = [x_0, x_1, x_2, ... x_{K-1}] \) is the corresponding input vector with \( K \) entries.

Consider each entry is expressed in binary encoded bits as follows:

\[
x_k = \sum_{n=0}^{N-1} b_{kn} 2^n, b_{kn} \in [0,1] \quad (2)
\]

where \( b_{kn} \) is the \( n \)th bit of \( x_k \) and has binary value either 0 or 1, while \( 2^n \) denotes the associated weight of binary bit.

Substituting equation (2) in (1) we get

\[
y = \sum_{k=0}^{K-1} C_k x_k = \sum_{k=0}^{K-1} C_k \sum_{n=0}^{N-1} b_{kn} 2^n \quad (3)
\]

By rearranging the equation (3) as follows:

\[
y = \sum_{n=0}^{N-1} \left\{ \sum_{k=0}^{K-1} C_k b_{kn} \right\} 2^n \quad (4)
\]

\[
f(C_k, b_{kn}) = \sum_{k=0}^{K-1} C_k b_{kn} \quad (5)
\]

Let

\[
y = \sum_{n=0}^{N-1} f(C_k, b_{kn}) 2^n \quad (6)
\]

The function \( f(C_k, b_{kn}) \) represents the sum of products and these values stored in the LUT. The DA-LUT output is multiplied with individual scaling factor and then added to the previous accumulated value [11].

**Unsigned and signed DA Schemes**

The basic use of the unsigned DA is to store the sum of the positive coefficients in the LUT as per input data samples. In each iteration cycle, the accumulator contents employ single bit shifter instead of the barrel shifter, to reduce the hardware cost. The implementation of the FIR filter with signed DA, requires the sign control accumulator for signed input values, and signed coefficients, as shown in Fig. 1. In this paper, an accumulator with addition/subtraction control is used to design a signed DA scheme, as it shifts the accumulator contents by a single shift in each iteration cycle. The memory address requirements get increased by applying the signed DA method using a ROM with single additional input due to the increase in the input bit requirements in the table.

Fig. 1. DA FIR filter.
Problem Formulation

The most important parameters in the FPGA based design applications are area utilization and operating frequency. The enhancement of these parameters is crucially required because of the increase in application complexities. Thus, various techniques and methods were proposed to effectively optimize the area and operating frequency. The proposed DA-RNS based scheme is fruitful for the implementation of high order FIR filters along with high operating frequency.

DA Based RNS FIR Filter

In this paper, single residue channel is used to implement the DA based RNS FIR filter which results in high operating frequency with minimum area usage as shown in Fig. 2. Through the application of RNS technique, the DA inner product equation can be expressed as follows:

\[ y_i = \left\lfloor \sum_{n=0}^{N-1} \left( \sum_{k=0}^{K-1} C_k b_{kn} \right) 2^n \right\rfloor_{m_i} \]

\[ = \sum_{n=0}^{N-1} f(C_k, b_{kn}) 2^n \right\rfloor_{m_i} \]  

(7)

The 2^n factor is required to get decoupled by utilizing the algebra of RNS as follows:

\[ y_i = \left\lfloor \sum_{n=0}^{N-1} f(C_k, b_{kn}) 2^n \right\rfloor_{m_i} \]  

(8)

\[ f_{m_i} (C_k, b_{kn}) = f(C_k, b_{kn}) 2^n \right\rfloor_{m_i} = \sum_{k=0}^{K-1} C_k b_{kn} \]  

(9)

By substituting (9) in (8) we get,

\[ y_i = \left\lfloor \sum_{n=0}^{N-1} f_{m_i} (C_k, b_{kn}) 2^n \right\rfloor_{m_i} \]  

(10)

The function \( f_{m_i} (C_k, b_{kn}) \) in the equation (10) shows that the data contents of LUT are stored in the form of residue number system. Then, the result of the look up table is multiplied with the individual modulo scaling factor and is then added to the value accumulated previously.
Simulation Based Results

The proposed scheme outperforms the state of the art approaches in terms of area utilization and operating frequency as shown in the Table I.

**TABLE I. RESOURCE UTILIZATION ANALYSIS**

<table>
<thead>
<tr>
<th>Logic Utilization</th>
<th>RAG based FIR Filter</th>
<th>Signed DA FIR Filter</th>
<th>Unsigned DA FIR Filter</th>
<th>Proposed Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Slice Registers</td>
<td>29</td>
<td>43</td>
<td>42</td>
<td>23</td>
</tr>
<tr>
<td>Number of Slice LUTs</td>
<td>43</td>
<td>56</td>
<td>56</td>
<td>24</td>
</tr>
<tr>
<td>Number of fully used LUT-FF pairs</td>
<td>20</td>
<td>38</td>
<td>42</td>
<td>23</td>
</tr>
<tr>
<td>Number of bonded IOBs</td>
<td>17</td>
<td>41</td>
<td>39</td>
<td>24</td>
</tr>
</tbody>
</table>

REFERENCES