Area – Efficient VLSI Implementation for Parallel Linear-Phase FIR digital Filters

S. Rajeswari¹, S. Suresh²

¹Department of ECE, Sri Ganesh college of Engineering & technology. ²Dr. Pauls engineering college

Abstract

The fast finite-impulse response (FIR) algorithms (FFAs), able to create a new parallel FIR filter structures, which are beneficial to symmetric coefficients in terms of the hardware cost, under the condition that the number of taps is a multiple of 2 or 3. The parallel FIR structures exploit the inherent nature of symmetric coefficients reducing half the number of multipliers in sub filter section at the expense of additional adders in pre-processing and post processing blocks. Exchanging multipliers with adders is advantageous because adders weigh less than multipliers in terms of silicon area; in addition, the overhead from the additional adders in pre-processing and post processing blocks stay fixed and do not increase along with the length of the FIR filter, whereas the number of reduced multipliers increases along with the length of the FIR filter. In the proposed work by mathematical and stimulation results it is shown by using Iterated Short Convolution (ISC) based linear convolution structure is transposed to obtain a new hardware efficient FFA filter structure which reduce multipliers in terms of hardware cost, especially when the length of the FIR filter is large.

Index Terms: Finite Impulse Response, Iterated Short Convolution, Digital Signal Processing, VLSI.

1. INTRODUCTION

The explosive growth of multimedia application, the demand for high-performance and low-power digital signal processing (DSP) is getting higher and higher. The FIR digital filter is one of the most widely used fundamental devices performed in DSP systems, ranging from wireless communications to video and image processing. Some applications need the FIR filter to operate at high frequencies such as video processing, whereas some other applications request high throughput with a low-power circuit such as multiple-input–multiple-output systems used in cellular wireless communication. Furthermore, when narrow transition band characteristics are required, the much higher order in the FIR filter is unavoidable. In this brief, parallel processing in the digital FIR filter will be discussed. Due to its linear increase in the hardware implementation cost brought by the increase in the block size L, the parallel processing technique loses its advantage to be employed in practice.

To reduce the complexity of the parallel FIR filter polyphase decomposition is mainly manipulated, where the small-sized parallel FIR filter structures are derived first and then the larger block-sized ones can be constructed by cascading or by iterating small-sized parallel FIR filtering blocks. In the past they used fast FIR algorithms (FFAs) to design new parallel FIR filter architectures, which are beneficial to symmetric convolutions of odd length and even length in terms of the hardware cost. Exchanging multipliers with adders is advantageous because adders weigh less than multipliers in terms of silicon area, and in addition, the overhead from the increase in adders in pre-processing and post processing blocks stay fixed, not increasing along with the length of the FIR filter, whereas the number of reduced multipliers increases along with the length of the FIR filter. To reduce this complexity in this work they designed new parallel architectures based on iterated short convolution (ISC) which is able to reduce more number of multipliers. A brief introduction of FFAs is reviewed in (2). In Section (3), the iterated short convolution. In Section (4) comparisons are shown. Finally, the conclusion is given in Section (5).

2. FFA

Consider an N-tap FIR filter that can be expressed in the general form as

$$y(n) = \sum_{i=0}^{N-1} h(i)x(n-i)$$

Where x (n) is an infinite length input sequence and h (i) represents the length-N FIR filter coefficients. Then, the traditional L-parallel FIR filter can be derived using polyphase decomposition as [3]
2.1 Three parallel filter using FFA

\[
Y_0 = H_0X_0 - Z^{-3}H_1X_2 + Z^{-3}[(H_1 + H_2)(X_1 + X_2) - H_1X_1] \\
Y_1 = [(H_0 + H_1)(X_0 + X_1) - H_1X_1 + [H_0X_0 - Z^{-3}H_1X_1] \\
Y_2 = [(H_0 + H_1 + H_2)(X_0 + X_1 + X_2)] - [(H_0 + H_1)(X_0 + X_1) - H_1X_1] \\
- [(H_1 + H_2)(X_1 + X_2) - H_1X_1]
\]

2.2 Structures for symmetric convolution of odd length

To exploit the symmetry of coefficients, the main idea is to manipulate the polyphase decomposition to earn as many subfilter blocks as possible, which contain symmetric coefficients so that half the number of multipliers within a single sub filter block can be utilized for the multiplications of whole taps.

For a set of symmetric coefficients in odd length earn two more subfilter block.

Example:
Consider for 27 tap filter with the set of symmetric coefficients as
\{ h(0), h(2), h(3), h(4), ..., h(27) \}
\[
Y_0 = H_0X_0 - Z^{-3} \\
\times (H_1 + H_2)(X_1 + X_2) - H_1X_1 \\
- \left( (H_0 + H_2)(X_0 + X_2) - H_0X_0 \right) \\
- \frac{1}{2} \left( (H_0 + H_2)(X_0 - X_2) \right) \\
- (H_0 - H_2)(X_0 - X_2)) \\
\]

\[
Y_1 = (H_0 + H_1 + H_2)(X_0 + X_1 + X_2) - (H_0 + H_2)(X_0 + X_2) + (H_0 + H_2)(X_0 - X_2) - \frac{1}{2} \\
\times [(H_0 + H_2)(X_0 + X_2) - (H_0 + H_2)(X_0 - X_2)] \\
- (H_0 - H_2)(X_0 - X_2) - H_0X_0 \}
\]

\[
Y_2 = (H_0 + H_2)(X_0 + X_2) - (H_0 - H_2)(X_0 - X_2) \\
\times \frac{1}{2} \\
\times [(H_0 + H_2)(X_0 + X_2) - (H_0 - H_2)(X_0 - X_2)] \\
\]

2.3 Customized cascading FFA
The proposed cascading process for the larger block-sized proposed parallel FIR filter is similar to that introduced in [1], but instead of applying the existing small-sized FFAs to every stage, we interleave multiple various small-sized structures in each stage to fully exploit the symmetry of coefficients. For two-parallel-based cascading, with a set of odd-length symmetric coefficients, it is possible to have even-length symmetric coefficients in a subfilter block after applying two-parallel FFA structure. For example, for a set of 23-tap symmetric coefficients, after applying two-parallel FFA, the subfilter block $H_0$ is with 12 symmetric coefficients, to which the existing FFA is not beneficial. The resulted four-parallel filter realization, which leads to four subfilter blocks containing symmetric coefficients, is shown in Fig. 4.

3. ITERATED SHORT CONVOLUTION

The Iterated Short Convolution (ISC) based linear convolution structure is transposed to obtain a new hardware efficient FFA filter structure which saves a large amount of hardware cost, especially when the length of the FIR filter is large. i.e., larger block-sized filtering structures can be constructed through iterations of the small-sized filtering structures. However, in both categories, symmetry of coefficients in the filter design has not been taken into consideration for the design of structures yet. This can lead to significant savings in hardware complexity and cost. In this paper, we provide a new parallel FIR filter structure based on FFA consisting of advantageous poly phase decompositions, which can reduce amounts of bulky multiplications in the sub-filter section by exploiting the inherent nature of the symmetric coefficients compared to the existing FFA fast parallel FIR filter structure.

3.1 Fast parallel fir algorithm based (ISC)

The iterated convolution structure can be transposed to obtain a fast parallel FIR filter. An $L(L=L_1L_2...L_r)$ parallel N–tap FIR filter based on iterated $L_1 \times L_2$ convolutions

$$ S_{2L-1} = Q'_{L-1}H_{L}P_{L-1}X_{L} $$

where(i=0,1,2,...,r).

$$ Y_L = P'_{L}H_{L}Q'^T_{L}A^T_{L}X_{L} $$

$P'$ and $Q'$ are the preprocessing and post processing matrices, which determine the manner in which the inputs and outputs are combined. The computational efficiency of the proposed parallel FIR filter structures is dependent on that of the selected short convolution algorithms.

This algorithm is combined with fast two and three point convolution algorithms to obtain a general iterated short convolution algorithm (ISCA). Although fast convolution of any length can be derived from Cook–Toom algorithm or Winograd algorithm.

4. FPGA IMPLEMENTATION ANALYSIS

The existing and proposed FFA structures are implemented in Verilog HDL targeted on Xilinx Virtex E FPGA device of filter length 27, word length of 8-bit. The comparison results are tabulated as

<table>
<thead>
<tr>
<th>Length</th>
<th>Structure</th>
<th>Power</th>
<th>Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>27-TAP</td>
<td>FFA</td>
<td>22726</td>
<td>3.28</td>
</tr>
<tr>
<td></td>
<td>ISA</td>
<td>22356</td>
<td>2.857</td>
</tr>
</tbody>
</table>

4.1 Software implementation

Parallel FIR filter algorithm, the preprocessing, Post processing and sub filter matrices are calculated easily with Matlab. Then, Matlab is used to automatically generate Verilog code for the hardware implementation of this algorithm. This automation is very efficient when the filter coefficients, word length or level of parallelism changes, especially when the length of the FIR filter is large.
Fig 7. Simulink model 27-tap filter using FFA

Fig 7 shows the Simulink model of the 27-TAP filter designed using the Xilinx block set. The Xilinx block set consist of the Basic element tools in that we can have the adders, multipliers. The input and output cannot be directly given it is given through the gateway in and gateway out.

Fig 8. Simulink model 27-tap filter using ISA

The addition operation is implemented with the carry-save adder, which can be used to convert the sum of numbers into the sum of 2 numbers. For example, a 4-input carry-save adder can be implemented as shown in Fig. 6. Before CSD multiplication, coefficient quantization is performed using a look-ahead maximum absolute difference (MAD) algorithm [5]. After the look-ahead MAD quantization is used to quantize subfilter coefficients, a CSD multiplier is automatically generated according to the nonzero bits of the quantization result. During the CSD multiplication, Horner’s rule [4] is used to improve the computation accuracy. Carry-save adders are used to accumulate all shifted numbers in each tap of the subfilters; the number of inputs of the carry-save adder is the number of nonzero bits of the corresponding coefficient than two affiliations.

5. CONCLUSION

In this brief, we have presented new parallel FIR filter structures, which are beneficial to symmetric convolutions of odd length. Multipliers are the major portions in hardware consumption for the parallel FIR filter implementation. The proposed new structures exploit the nature of symmetric coefficients of odd length and further reduce the amount of multipliers required at the expense of additional adders. Since multipliers outweigh adders in hardware cost, it is profitable to exchange multipliers with adders. Moreover, the number of increased adders stays still when the length of FIR filter becomes large, whereas the number of reduced multipliers increases along with the length of the FIR filter to overcome this we using ISCA to reduce further multipliers.

REFERENCES