Implementation Of Efficient Fused Add-Multiply Operator Using Compressor

Anila M 1, Binu Manohar 2

1 2 Department of Electronics and Communication Engineering
1 Mangalam College of engineering
2 Ettumanoor

Email- anilamukundhan@gmail.com

Abstract- Many Digital Signal Processing (DSP) applications carry out a large number of complex arithmetic operations. Multiplier take important role in high performance of the system, reduce in power and delay. This paper is focus on optimizing the design of Fused Add Multiply (FAM) operator. To increase the performance and to reduce the complexity of arithmetic operations, we designed a Fused Add-Multiply operator which directly recodes the sum of two numbers in its modified booth (MB) form and uses 4:2 compressor for the partial product addition. The proposed technique focus on FAM design by using compressor at the last stage of partial product addition which reduces both power consumption, area and delay leading to more efficient design for high performance applications.

Keywords – Add-Multiply operation, Modified Booth recoding, S-MB recoding schemes, 4:2 Compressor

I.INTRODUCTION

In many digital signal processing (DSP) and multimedia applications, multiplication and addition are the Most commonly used operations. Therefore, multiply-add fused unit plays an important role in improving performance by combining multiplication and addition operation into a single unit in the modern embedded processors. For this, a specially designed signed bit adders are used. Here, we use a radix-4 modified booth’s algorithm to increase the speed of multiplication. As the radix size increases, the number of partial products are reduced, which uses small adder tree for the addition of partial products, which in turn increases speed and reduces area. The fused operation is mainly performed in three steps: S-MB recoding, a Wallace tree, and a final adder. In the existing system carry look ahead or ripple carry adder is used at the last stage. Here, we replace it with compressor which reduces delay, area and power as the addition is performed in parallel.

The rest of paper is organized as follows. The existing design scheme is explained in section II. The proposed S-MB recoding scheme is explained in section III. Experimental results are presented in section III. Concluding remarks are given in section IV.

II. EXISTING FAM

In conventional design of AM operator addition and multiplication are performed separately. First addition is done by using any conventional adder and then the input and the result of adder are driven to a multiplier to get the output. The disadvantage of using a separate conventional adder is that it inserts a significant critical path delay. As the bit width increases critical path delay also increases, because the carry signals are to be propagated inside the adder. To overcome this we used a Carry Look Ahead (CLA) adder, which however increases area and power dissipation. In order to reduce this area and power dissipation, an optimized design of the AM operator which combines the adder and MB encoding unit into a single data path block by direct recoding the sum of two numbers into its modified Booth form as in figure 1.
Implementation Of Efficient Fused Add-Multiply Operator Using Compressor

![Diagram of Existing FAM](image1)

**Figure 1: Existing FAM**

The fused Add Multiply (FAM) operator uses only one adder at the last stage of addition; as a result it reduces both area and critical path delay. A way to increase the speed of multipliers is through the use of Modified Booth algorithm to generate the partial products. The generated partial products are added by using Wallace tree instead of normal full adders. The carry look ahead adder is performed for the sum and carry bit generated from the Wallace CSA at the last stage for the final result.

**III. PROPOSED FAM**

This paper focuses on the design and implementation of the fused AM unit which implements the operation $Z=X \cdot (A+B)$. The compressor is used at the last stage of fused Add-Multiply (FAM).

![Diagram of FAM with Compressor](image2)

**Figure 2: FAM with Compressor**
A. S-MB recoding schemes–

In this recode the sum of two consecutive bits of the input A with two consecutive bits of the input B into one MB digit yjMB. S-MB recoding is mainly three different schemes so a set of bit level Half Adders (HAs) and Full Adders (FAs) are developed for recoding scheme.

1) S-MB1 Recoding

This recoding scheme, we used conventional and signed FA. Which is used for both odd and even number of bit width of input numbers. For generating the MB digit yjMB, 0 ≤ j ≤ k-1 we need 3 bits (s2j+1, s2j, c2j) which are outputs of jth recoding cell having the inputs a2j, a2j+1 and b2j, b2j+1. The sum bits s2j+1 and s2j are generated from the jth recoding cell and c2j bit is generated from the conventional FA, which have a2j-1, b2j-1 and b2j-2 as inputs.

2) S-MB2 Recoding

This recoding scheme used signed FA and HA. Which is used for both odd and even number of bit width of input numbers applied. Initially we consider c0, 1 = c0, 2 = 0. For form the MB digit yjMB, 0 ≤ j ≤ k-1 we need three bits (s2j+1, s2j, c2j, 2) which are the outputs of jth recoding cell. As in the S-MB1 recoding scheme, we use a FA to produce the sum s2j and the carry c2j+1 with a2j, b2j, c2j, 1 as inputs. The bit c2j, 1 is the output carry of the conventional HA of the previous recoding cell and has the bits a2j-1, b2j-1 as inputs. The HA* which is negatively signed produces the output bit s2j+1.

3) S-MB3 Recoding

In this recoding scheme, use a conventional FA, signed HA and FA. First we consider c0, 1 = c0, 2 = 0. As in the previous schemes, we use a FA to produce sum s2j and the carry c2j+1 with a2j, b2j, c2j, 1 as inputs. The bit c2j, 1 is the output carry of the signed HA (HA*) of the previous recoding cell and from HA** which is negatively signed produces output bit s2j+1.

B. Radix-4 Booth Recoding –

For high speed operation here uses radix4 recoding. The multiplier term is recoded. We consider the bits in blocks of three. Grouping of bits starts from the LSB. The overlap is necessary so that we know what happened in the last block, as the MSB of the block act like a sign bit. The first block only uses two bits of the multiplier and assumes a zero for the third bit. Then compare the bits with the booth recoding table shown in table 1 to generate partial products. By using S-MB recoding schemes, we convert the sum of two consecutive bits of two inputs into three bits and then obtained bits are compared with the booth table to generate partial products. After the partial products are generated, they are added with compressor along with the correction bits for generating the result.

Figure3: Grouping of bits used in Booth recoding.

C. 4:2 Compressor–

The 4:2 compressor design reduces the complexity of the Wallace tree structure for multiplication. A 4:2 compressor has four inputs and two outputs i.e. the sum and the carry apart from a carry in (cin) and carry out (cout)
Cin is the cout of the previous compressor and cout becomes the cin of the next compressor. The block structure for a 4:2 compressor can be realized as shown in figure.

![4:2 compressor](image)

The standard governing equation for a 4:2 compressor can be considered as follows:

$$a_0 + a_1 + a_2 + a_3 + a_4 + cin = sum + 2^4 (\text{carry} + cout)$$

IV. EXPERIMENT AND RESULT

The proposed Fused add unit with three different recoding scheme is implemented in the VHDL, simulated with Xilinx and modelsim. The three recoding schemes are implemented in VHDL for both cases of even and odd bit-width of the input numbers. The lowest area, delay and power values are marked.

A. Simulations

The three recoding schemes are implemented in VHDL for both cases of even and odd bit-width of the input numbers. Xilinx is used for simulation.

![Simulation result](image)
B. Performance analysis

The three different schemes are compared in terms of PDP and ADP consumption is given below.

<table>
<thead>
<tr>
<th>Recoding Schemes</th>
<th>Existing scheme PDP (X10^-12)</th>
<th>Proposed scheme PDP (X10^-12)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Even</td>
<td>Odd</td>
</tr>
<tr>
<td>S-MB2</td>
<td>10.304</td>
<td>11.302</td>
</tr>
</tbody>
</table>
Table 2: Area delay product comparison of the recoding scheme

<table>
<thead>
<tr>
<th>Recoding Schemes</th>
<th>Existing scheme ADP (X10^4)</th>
<th>Proposed scheme ADP (X10^4)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Even</td>
<td>Odd</td>
</tr>
<tr>
<td>S-MB1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.620</td>
<td>5.482</td>
</tr>
<tr>
<td>S-MB2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.876</td>
<td>5.825</td>
</tr>
<tr>
<td>S-MB3</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.981</td>
<td>5.554</td>
</tr>
</tbody>
</table>

V. CONCLUSION

The design of sum-product are used to implement the direct recoding of the sum of two numbers in its Modified Booth (MB) form to synthesize each Fused sum-product design. The proposed recoding schemes (S-MB1, S-MB2 and SMB3) achieve delay and power has been reduced. The performance of the Fused sum-product designs that include the proposed recoding schemes is considered with respect to the bit width of the input numbers. The delay and power measurements of all Fused sum-product units for even and odd bit-width of the recoder. The use of compressor helps to improve performance of FAM unit.

REFERENCE