

Comparison of different multiplier algorithms and 1D-DWT as an application

Athira Koranath, Sonali Agrawal

Amrita University

AMRITA SCHOOL OF ENGINEERING, Kasavanahalli, Carmelaram P.O.

Bangalore - 560 035

ABSTRACT : This paper compares 4 different multipliers and the one with the least area and power is applied in 1D-DWT. The multipliers are implemented using booth algorithms as well as Wallace tree structures. The advantage of using Modified Booth algorithm is that the number of partial products is reduced by half. The Wallace tree structure is used for partial product reduction. 1D-DWT is implemented using modified Wallace tree structures alone as it has the least area and power. The 1D-DWT is mainly used for image compression.

Keywords – Modified Booth algorithm, Wallace trees, 1D-DWT.

I. INTRODUCTION

Multipliers form a major part of DSP applications. This paper focuses on different combinations of multipliers using Wallace trees and modified booth algorithm. The multipliers are used in 1D-DWT which is used for image compression and signal processing applications.

Any multiplier design has 3 steps. They are i) partial product generation ii) partial product reduction iii) final addition. The partial products are formed first either by using modified booth algorithm or by ANDing each bit of the multiplier with each bit of the multiplicand. The next step is reduction of these partial products to two rows. This is done by grouping the partial products into groups of three. These groups are reduced by using carry save adders. Carry save adders are mainly full adders and half adders. The third step is addition of the final two rows by using carry look ahead adders to yield the final product.

This paper focuses on 4 different combinations of multipliers. They are i) conventional Wallace trees ii) modified Wallace trees iii) modified booth- Wallace trees iv) modified booth- modified Wallace trees. They are compared in terms of area, power and delay. It is seen that modified Wallace tree has the least area and power. Hence the modified Wallace tree is used for 1D-DWT. The DWT is also used for signal processing applications. 1D-DWT uses low pass and high pass filters for signal analysis and reconstruction.

The area , power and delay parameters are also found out using synopsys design compiler and Xilinx tools. The area and power are also listed in the tables in section IV.

II. DESIGN APPROACH

This section describes about the design of the different multipliers.

2.1 Conventional Wallace trees

A Wallace tree is an efficient hardware implementation of a digital circuit that multiplies two integers. The Wallace tree is used for reduction of partial products. The first step is ANDing of each bit of the multiplier with each bit of the multiplicand to yield the partial product. The next step is reduction of partial products to two rows. This is done by grouping the partial products to rows of three. There are 3 rules in partial product reduction. i) apply a full adder to each column which contains 3 bits ii) apply a half adder to each column which contains 2 bits iii) a single bit column is passed on as such. After the reduction we get two rows. These two rows are added using carry save adders.

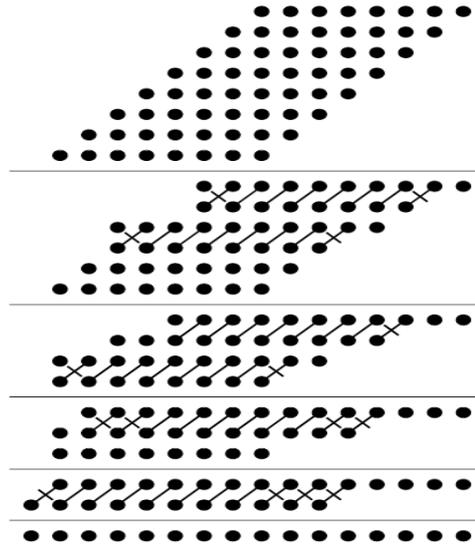


Fig 1. 8 bit conventional wallaec tree

The number of partial products is determined in each stage is determined by the following equation:

$$W_0 = N$$

$$W_{j+1} = 2 \lfloor W_j / 3 \rfloor + W_j \bmod 3$$

2.2 Modified Wallace tree

In this multiplier we first form an inverted pyramid array by rearranging the bits in the partial product array.

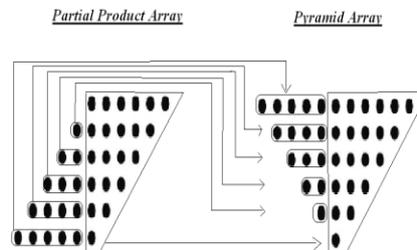


Fig 2. Conversion of partial product array to pyramid array

The next step is partial product reduction. This is done again by grouping the partial product array into 3 rows. In this case we don't use the half adders as we did for conventional Wallace tree. So the number of half adders is reduced. Here we use half adders whenever the number of rows in each stage exceeds that of the previous stage.

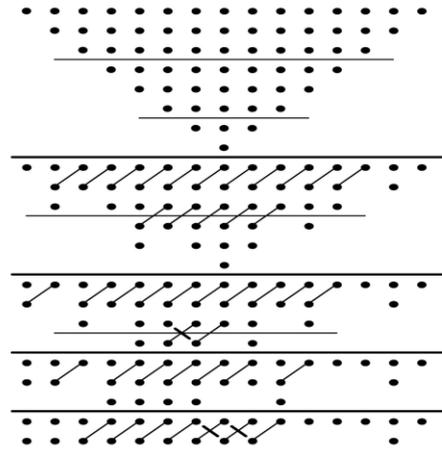


Fig 3. 8bit Modified Wallace tree

2.3 Modified Booth-Wallace tree

This multiplier uses both Modified Booth algorithm and conventional Wallace tree structures. Modified Booth algorithm is used for partial product generation as it reduces the number of partial products by half. There are six steps for partial product formation. They are:

1. Pad the LSB with one zero.
2. Divide the multiplier into overlapping groups of 3-bits.
3. Determine partial product scale factor from modified booth 2 encoding table.
4. Compute the Multiplicand Multiples
5. Sum Partial Products

Table 1. Booth recoding table

| Groups | Operation |
|--------|------------------------|
| 000 | All zero's |
| 001 | '0' input values in |
| 010 | '0' input values in |
| 011 | Shift |
| 100 | 2's complement + shift |
| 101 | 2's complement |
| 110 | 2's complement |
| 111 | All zeros |

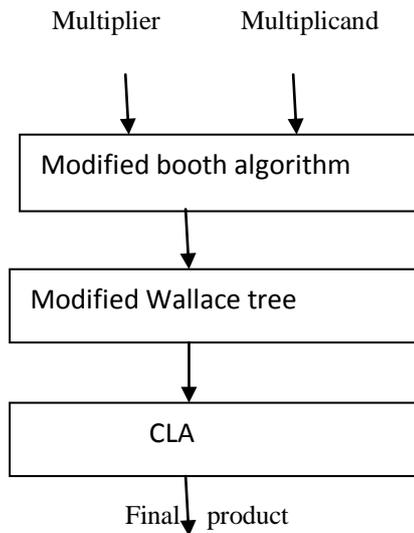


Fig 4. Block diagram

2.3 Modified Booth-Modified Wallace tree

This multiplier uses modified booth algorithm for partial product generation and modified Wallace tree structures for reduction of partial products. The final addition is carried out by carry look ahead adders.

III. 1D-DWT AS APPLICATION

1D-DWT stands for one dimensional discrete wavelet transform. It is a transform similar to discrete fourier transform(DFT). It uses multi-resolution technique for time-frequency analysis of signals. The main advantage of DWT compared to DFT is that we get both time and frequency analysis of a signal at the same time.

The 1D-DWT is mainly used for image compression. It also has various signal processing applications. The DWT can provide significant compression ratios than the previous techniques like the Discrete Cosine Transform(DCT) and the Discrete Fourier Transform(DFT).

The signal to be analysed is passed through low pass and high pass filters. This is then followed by decimation by two. This yields the low pass sub-band y_L and the high pass sub-band y_H . To reconstruct the original signal we first do interpolation followed by low pass and high pass filtering. This is illustrated in the figure below.

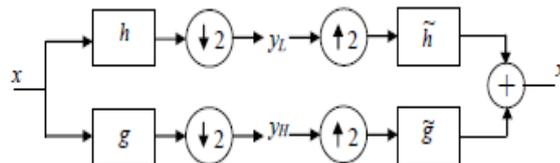


Fig 5. Signal analysis and reconstruction in DWT

The method used for finding the DWT convolution based method. The DWT is found out for a (9, 7) filter. This filter has 9 low pass and 7 high pass coefficients. In the figure D represents the delay elements. Also in this instead of normal multipliers we use modified Wallace trees as it has the least area and power when compared with the other three multipliers

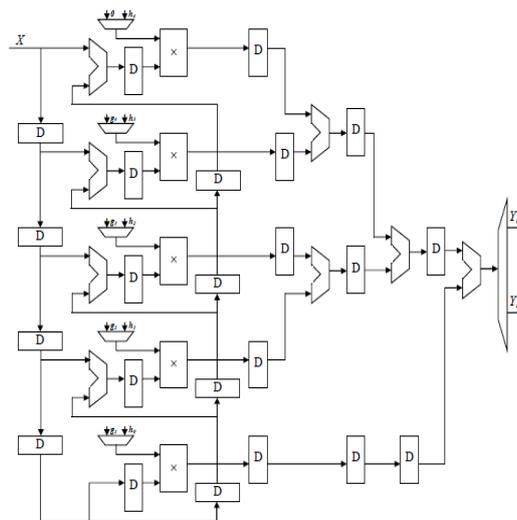


Fig 6. 1D-DWT for (9,7) filter

IV. RESULTS

4.1 Simulation results

The simulations were carried out in Model Sim 6.4. the simulations were carried out for 8 bit Wallace tree, modified Wallace tree, modified booth-Wallace tree and modified booth-modified Wallace tree. The implementation was done using VHDL and results were checked for different values.

4.2 Area and Power

The area and power were reported using Synopsys design compiler.

Table 2. Area – Power comparison

| Multiplier | Area | Power |
|--------------------------------------|-------------|------------|
| Wallace tree | 3568.271685 | 339.6556uw |
| Modified booth-Wallace tree | 5103.555954 | 288.2324uw |
| Modified Wallace tree | 2996.965206 | 99.5429uw |
| Modified booth-Modified Wallace tree | 5096.312423 | 288.4743uw |
| 1D-DWT | 22,419.8647 | 977.3285uw |

4.3 Delay calculation

The delay was calculated using Xilinx ISE.

Table 3. Delay calculation

| Multiplier | Delay |
|--------------------------------------|---------|
| Wallace tree | 21ns |
| Modified booth- Wallace tree | 8.413ns |
| Modified Wallace tree | 27ns |
| Modified booth-modified Wallace tree | 8.413ns |

The results show that the area and power is least for modified Wallace tree but delay is maximum for it. So there is a trade off between area , power and delay.

V. CONCLUSION

This project compares 4 different multipliers and it was found that the modified Wallace tree is the best in terms of area and power. Hence the modified Wallace tree was used to implement the 1D-DWT for (9,7) filter coefficients and the results were obtained.

REFERENCES

- [1] Lakshmanan,M.othman and M.A.Mohd.Ali, "High performance parallel multiplier using Wallace-booth algorithm", 2002,proceedings ICSE 2002,IEEE International conference on Semiconductor Electronics
- [2] Ron S. Waters and Earl E. Swartzlander, A Reduced Complexity Wallace Multiplier Reduction, *IEEE TRANSACTIONS ON COMPUTERS*, VOL. 59, NO. 8, AUGUST 2010
- [3] Karen, *Computer Arithmetic Algorithms*. Prentice Hall. 1993. S.Y. Kung, VLSI array processors, Prentice Hall. 1998.
- [4] A.D. Booth, A signed binary multiplication technique, *Quart. J. Mech. Appl. Math.*, vol.1.4 pp 236-240, 1951
- [5] L.P. Rubinfield, A Proof of the modified Booth algorithm for multiplication, *IEEE Trans on Computers* C-24 (Oct-1975)
- [6] David A. Patterson & John L.Hennessy, *Computer Architecture-A Quantitative Approach*. Morgan Kaufmann, 1996.
- [7] C.S. Wallace, A suggestion for fast multipliers, *IEEE Trans. Electronics. Comput.*,vol.EC-13,pp.14-1F7e,b .1964
- [8] S.Shah, A.J. Al-kalili, D.Al-khalili, "comparison of 32-bit Multipliers for various performance measures", 2000.ICM2000, *Proceedings of the 12th International Conference on Microelectronics*
- [9] Rizalafande Che Ismail , R.Hussin, "High performance complex number multiplier using Booth-Wallace algorithm", 2006,ICS2006,IEEE International Conference on Semiconductor Electronics
- [10] Soojin Kim and Kyeongsoon Cho, "Design of high-speed modified booth multipliers operating at GHz ranges", *World Academy of Science, Engineering and Technology* 61 2010
- [11] Manish Bansal, Sangeeta Nakhate, Ajay Somkuwar, "High performance pipelined 64*64 bit multiplier using radix-32 modified booth algorithm and Wallace structure", 2011 International conference on computational intelligence and communication networks
- [12] Mountassar Maamoun, Mehdi Neggazi, Abdelhamid Meraghni, and Daoud Berkani, " VLSI design of 2D Discrete Wavelet Transform for Area-efficient and High-Speed Image Computing", *Proceedings of World Academy of Science, Engineering and Technology* Volume 35 November 2008 ISSN 2070-3740.