

Digital Signal Processing Applications using Multiplier Technique in Fixed Point Arithmetic

V. Indhumaraghavalli¹, S. Rajan²

¹ME-VLSI Design - Final Year, Department of ECE, Velalar College of Engineering and Technology, Erode, Tamil Nadu, India

²Assistant Professor (Sr. Gr.), Department of ECE, Velalar College of Engineering and Technology, Erode, Tamil Nadu, India

Abstract: *Embedded and mobile computing devices are frequently required to execute some key digital signal processing applications. Such applications using fixed point arithmetic for some computational errors. Most of the digital signal processing application required to increase the speed. In this proposed multipliers are array multipliers, Booth multipliers and Modified booth multipliers compared with their power, area and accuracy. Finally, note that the modified booth multiplier is improving the performance based on the power area and accuracy. But the delay is reduced in the Booth multiplier. The computation error is present in the multipliers, but it does not affect the quality of the DSP applications.*

Keywords: fixed point, multipliers, multiplexers, truncation, power consumption

1. Introduction

Recent years have witnessed tremendous advancement of portable electronic devices powered by batteries with intensive computational capabilities. The power requirements of these devices have increased many folds with the increase in complexity of ICs.

Multiplication is one of the basic arithmetic operations but it requires more hardware sources. Advanced technology researchers, tried by using the multipliers to increase the speed and reduce the power in the digital signal processing applications. To determine the performance of the multiplier number of partial products to be added is the main parameter. Based on the application serial or parallel multiplier is selected.

The serial parallel multiplier achieves better performance by compromising the speed. But performing arithmetic operation, particularly multiplication operations, a typical processor central processing unit devotes a considerable amount of processing time. DSP system performance is limited by its multiplication performance. Multiplication dominates the execution time of most DSP algorithms.

Multiplications have always been a hardware, time and power consuming arithmetic operation, especially for large value operands. This bottleneck is even more emphasized in digital signal processing applications that involve a huge number of multiplications [4]. Achieving high energy efficiency has become a key design objective for embedded and mobile computing devices due to their limited battery capacity and power budget.

Parallel multipliers provide high speed method for multiplications, but require large area, for VLSI implementation. In most digital signal processing applications rounded product is required to avoid growth in word size. Thus an important aim is to design a multiplier which required less area and that is possible with the truncated multipliers. This multipliers are extensively used in digital signal processing where the speed of the

multiplication and the area and power consumption are important [8].

The fixed point arithmetic is used due to the area and cost of supporting FP units in embedded computing devices. Although this conversion process leads to some loss of computational accuracy, but it does not affect the quality of digital signal processing applications due to computational error tolerance. The approximate multiplication technique takes m-consecutive bits from each n-bit operand [1].

In this paper, proposes an approximate multipliers uses m bit segment from n bit operand. This approach provides higher accuracy in the LSB by truncating method. Because, it consider accurate all bits. Here compare the array, booth and modified booth multiplier based on the delay and power.

Rest of this brief explains the follows. Section II explains the existing method for the proposed architecture. Section III analyzes the proposed methods architecture operation and performance analysis of the multipliers based on quality of delay and accuracy. Section IV results and discussion of this paper and section V concludes the paper.

2. Existing Method

Digital signal processing algorithms often rely heavily on a large number of multiplications, which is both time and power consuming. Various types of multipliers are used to improve the accuracy and energy efficiency. An 16*16 multiplier proposed using inaccurate 2*2 partial product generators while guaranteeing the minimum and maximum accuracy fixed at design time.

2.1 Approximate Array Multiplier

Array multiplier is well known due to its regular structure. Multiplier circuit is based on add and shift algorithm. Each partial product is generated by the multiplication of the multiplicand with one multiplier bit. The partial product are shifted according to their bit orders and then added. The adders can be performed with normal carry propagate adder.

Both multiplicand and multiplier may be positive or negative, 2's complement number system is used to represent them. If the multiplier operand is positive then essentially the same technique can be used but care must be taken for sign bit extension.

To motivate and describe this multiplier, define an m-bit segment as m contiguous bits starting with the leading one in an n-bit positive operand. This method dub into dynamic segment method and in contrast to static segment method. With two m-bit segments from two n-bit operands, perform multiplication using an m*m multiplier. The simple example of a multiplication after taking 8-b segments from 16-b operands.

$$\begin{array}{r}
 * \quad \quad \quad 111\ 0111\ 0101\ 1010 \\
 \hline
 \quad \quad \quad 0000\ 0011\ 1011\ 1011 \\
 \hline
 = 0001\ 0001\ 0100\ 1001\ 0101\ 0000\ 0000\ 0000
 \end{array}$$

Figure 1: Example of multiplication with 8-b segments

Furthermore, an m*m multiplier consumes much less energy than n*n multiplier, because the complexity of multipliers quadratically increases with n. For example, the 4*4 and 8*8 multipliers consume almost 20% and 5% less energy than a 16*16 multiplier per operation on average.

The area and energy penalties associated with DSM are to capture an m-bit segment starting from an arbitrary bit position in an n-bit operand because the leading one bit can be anywhere. To overcome this SSM method can be used or multiplication, the m-bit segment that contains the leading one bit of each operand and apply the chosen segments from both operands to the m*m multiplier. The SSM greatly simplifies the circuits circuit that chooses m-bit segments and steers them to m*m multiplier by replacing two n-bit LODs and shifters for the DSM.

The accuracy of an SSM with m=n/2 can be significantly low for operands. On the other hand, this problem becomes less severe as m is larger than n/2. An SSM allowing taking an m-bit segment from two possible bit positions of an n-bit operand. The advantage is scalability for various m and n, because the complexity of circuits choosing/steering m-bit segments and expanding a 2m-bit result to a 2n bit results scales linearly with m.

The approximate multiplier takes m consecutive bits of an n-bit operand either starting from the MSB or ending at the LSB and apply two segments that includes the leading ones from two operands to an m*m multiplier. Compared with an approach that identifies the exact leading one positions of two operands and applies two m-bit segments starting from the leading one positions, consumes much less energy and area efficiency.

This improved energy and area efficiency comes at the cost of slightly lower compute accuracy. The loss of small compute accuracy using SSM does not impact QoC of image, audio and recognition applications.

3. Proposed Method

Embedded and mobile computing devices are frequently required to execute digital signal processing (DSP). Now days the embedded system sacrifices the power for speed. Speed is the major concern in the digital signal processing. The speed of the booth multiplier and modified booth multiplier is considered and it can be compare with the array multiplier.

The multiplicand and product are specified in this multiplier. Both also in two's complement representation, but any number system that supports addition and subtraction will work as well like multiplier. These typically proceeds from LSB and MSB starting at i=0; based on this the booth and modified booth multiplier works.

3.1. Booth Multiplier

The multiplier needs to reduce the number of partial products generated. The booth recording multiplier is like that multiplier; it uses three bits at a time to reduce the number of partial products. These 3 bits are: the 2 bit from the present pair and a third bit from the high order bit of an adjacent lower order pair.

To speed up the multiplication booth encoding performs several steps of multiplication at once. An adder subtractor is nearly as fast and small as a simple adder. This advantage is taken by Booth's algorithm.

If the successive bits in the multiplicand are same then the addition/ subtraction operation can be skipped in the booth multiplier. The delay and the adders are reduced in this multiplier. The power consumption is increased in the booth multiplier. It is one of the drawbacks in this type multiplier. The complexity also reduced due to the reduction of adder circuit.

The booth multiplication is a powerful algorithm for signed multiplication. It considers both positive and negative numbers randomly. This algorithm will reduce the number of multiplicand multiples.

For the standard add-shift operation, each multiplier bit generates one multiple of the multiplicand to be added to the partial product. Large number of multiplicands has to be added when the multiplier is large. The performance will get better, if there is away to reduce the number of addition.

3.2. Modified Booth Multiplier

The modified booth multiplier consists of three steps: 1. to generate partial products 2. to add the generated partial products until the least two rows are remained 3. to compute the final multiplication by adding the last two rows. It reduces the partial half products in the first step. It is noted as most efficient booth encoding and decoding scheme.

The modified booth algorithm is mainly used for high speed multiplier circuits. This modified booth multiplier's computation time and the logarithm of the word length of operands are proportional to each other. It reduces half the number of partial product.

The overlapping is used for comparing three bits at a time. This grouping is started from least significant bit(LSB), in which only two bits of the booth multiplier are used by the first block and a zero is assumed as third bit. It generates irregular partial product array.

The area is reduced due to its silicon area savings and the speed is increased as the number of stages reduced by half. The speed of the multiplier is mainly based on the partial product generate and how can add fast them.

To reduce the number of partial products to be added the modified booth algorithm is used. The power consumption also reduced significantly in the modified booth algorithm. It reduces the power at the same time the area and delay also reduced in the modified booth multiplier.

3.3 Operation of the Multipliers

Fig 2. Shows an SSM allowing taking an m-bit segment from two possible bit positions of an n-bit operand. The accuracy of the SSM method depends $m=n/2$. In this method two separate process used for MSB and LSB operation. Two 16 bit mux is used in the input side with the OR gate and it can be given to the multiplier and the half adder. The adder performs based on the requirement of the application. The adder is used based on the multiplier technique. The booth multiplier and the modified booth multiplier reduces the adders. Because of this the hardware complexity is reduced in this two multiplier compared with array multiplier. Then this multiplier and half adder output is given to the 3-to-1 multiplexer. The output of this multiplexer is 32 bit. The input is given as 16 bit for the MSB and LSB. Based on the truncation method the LSB considers and the accuracy is increased.

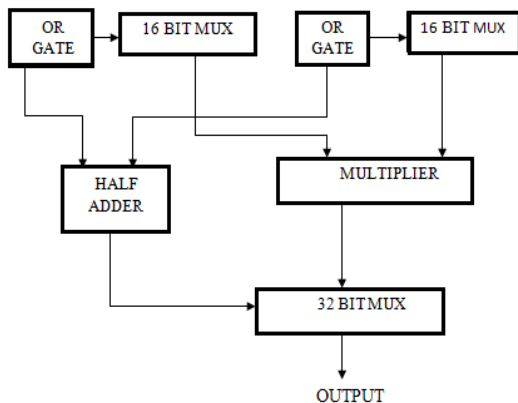


Figure 2: Architecture of the proposed method

One of operands for each multiplication is fixed co-efficient. In the proposed method to precompute the bitwise OR value of $B[n-1:m]$ and preselect between two possible m-bit segments (i.e., $B[n-1:n-m]$ and $B[m-1:0]$). Then it is store them instead of the native B value in memory.

This architecture support three possible starting bit positions for picking an m-bit segment where $m=n/2$. The two 2-to-1 multiplexers used at the input stage and one 3-to-1 multiplier at the output stage are replaced. This replacing is done with 3-to-1 and 5-to-1 multiplexers respectively. This can be

processed by making some minor changes in logic functions generating multiplexer control signals.

The evaluation of the computational accuracy and energy consumption of various multipliers are analysed. The VHDL code is used for verifying the output and power and delay are measured using the Xilinx software.

For computational accuracy evaluation, four sets of 16-b operand pairs from: 1) all possible pairs of 16-b values are taken in random manner; 2)by using audio the noise can be cancelled; 3)by image 2-D optical tomography; 4)by recognition isolated spoken digit recognition.

To evaluate delay and power the Xilinx software used. The two different multipliers are used like booth and modified booth multiplier. The booth multiplier is superior to all like speed, power and area. Modified booth sacrifices the speed for power. It reduces the power at the same time delay is increased compare with booth multiplier.

The area also reduced due to the reduction of the adder circuit in the architecture. Both of the multiplier reduces the adder operation. The complexity is also reduced in these multipliers. Then the performances are compared with the array multiplier.

Array multiplier is reduces the power and at the same time the speed and complexity is increased. The accuracy is small. The array multiplier gives the 99% computational accuracy with negligible noise in the image.

4. Result Analysis

The delay, power and area of the array multiplier, booth multiplier and modified booth multiplier are analysed. This can be done by using the ModelSim. Compared with these three multipliers the modified booth multiplier reduces the area, power and delay. Table1 shows the comparison of the three multipliers based on the power and delay.

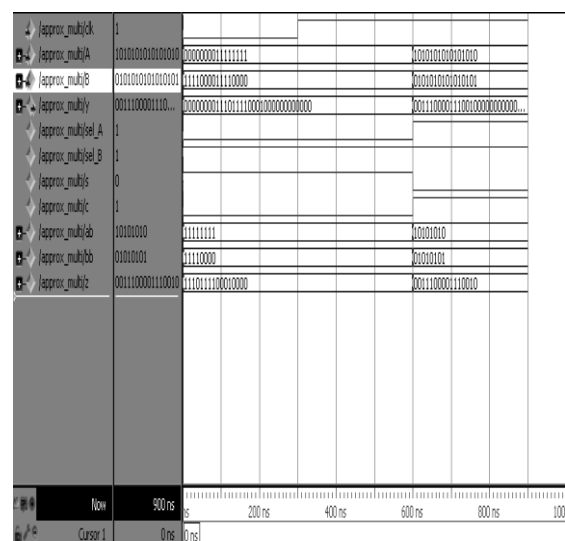


Figure 3: Output waveform for Array Multiplier

Fig.3 shows the output waveform of array multiplier. This is an simple multiplier output with better accuracy. The delay is reduced in this method so the speed is increased and the

power consumption is reduced. This is an 2 input multiplier. It's output used the MSB bit or LSB bit only. Because of this the accuracy is increased.

Table 1: Comparison of three multipliers

Parameter	Array	Booth	Modified Booth
Power	343 μ W	325 μ W	304 μ W
Delay	42.73ns	32.337ns	28.77ns

Fig 4 shows the power comparison of the array multiplier, booth multiplier and modified booth multiplier. From the figure the array multiplier consumes more power compared with the booth and modified booth. At the same time the modified booth multiplier consumes less power compared with other two multipliers.

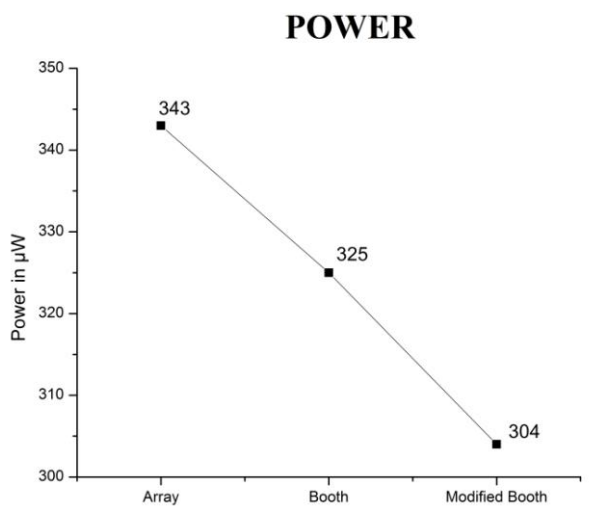


Figure 4: Power comparison

Fig.5 shows the delay comparison of the array multiplier, booth multiplier and modified booth multiplier. From the figure the array multiplier increases the delay compared with the booth and modified booth multiplier. At the same time the modified booth multiplier reduces the delay compared with other two multipliers.

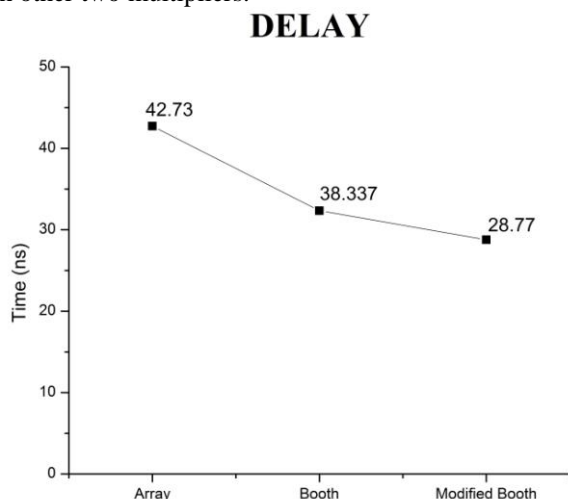


Figure 5: Delay comparison

5. Conclusion

In this paper Array multiplier, booth multiplier and modified booth multiplier were compared based on their delay and

power. The array multipliers power is increased and the delay also increased. Booth multiplier reduces the delay and power. The modified booth multiplier reduces the power at the same time the delay is also reduced compared with the two multiplier. In the DSP applications the speed is the major concern. Compared with these three multipliers the modified booth multiplier reduces the delay and power.

References

- [1] Srinivasan Narayanamoorthy, Hadi Asghari Moghaddan, Zhenhong Liu, Taejoon Park and Nam Sung Kim, "Energy-efficient approximate multiplication for digital signal processing and classification applications", IEEE transactions on very large scale integration (VLSI) systems, vol. 23.
- [2] Sumit Vaidya and Deepak Dandekar, "Delay-power Performance Comparison of Multipliers in VLSI circuit design", International Journal of Computer Networks and Communications (IJCNC), vol. 2, No.4, July 2010.
- [3] Suresh R. Rijal, Sharda G. Mungale, "Design And Implementation Of 8*8 Truncated Multiplier On FPGA", International Journal Of Scientific And Research Publications, Vol 3, Issue 3, March 2013.
- [4] Z. Babić, A. Avramović, and P. Bulić, "An iterative logarithmic multiplier." Microprocessors Microsyst., vol. 35, no. 1, pp. 23–33, 2011.
- [5] Manpreet Singh, Sukhmeet Kaur and suman, "Implementation of modified Booth algorithm (radix 4) and its comparison with booth algorithm (radix-2)", manna3issn 2231-1297, vol. 3, pp.683-690.
- [6] M. Ercegovic, P. Gupta and P. Kulkarni, "Trading Accuracy for Power with an Under Designed Multiplier Architecture", in proc. 24th IEEE Int. Conf. VLSI Design (VLSID), pp.346-351, Jan 2011.
- [7] Swapna M S and Vimala, "Delay Comparison of Various Multipliers With Adaptive Hold Logic (AHL)", International Journal of Advanced Research Trends in Engineering and Technology (IJARTET), Vol. 2. June 2015.
- [8] G.M. Nirmal, S. Rajan, "Power Optimization using Square Root Carry Select Adder", Transactions on Engineering and Sciences (TES), pp.318-321, 2015. ISSN: 2347-1964 (Online)
- [9] Jin U. Kang and Kang Zhang, "Speed Real-Time Fourier Domain Optical Coherence Tomography", IEEE journal of selected topics in quantum electronics, vol. 18, 2012.
- [10] Wang Z, Bovik A. C., Sheikh H. R., Simoncelli E. P., "Image quality assessment: From error visibility to structural similarity", IEEE Trans. Image Process., vol. 13, pp. 600–612. April 2004.
- [11] Yuan-Ho Chen, "An Accuracy Adjustment Fixed-Width Booth Multiplier Based on Multilevel Conditional Probability", IEEE transactions on very large scale integration (VLSI) systems, vol. 23., pp 1800-1805, jan 2015.
- [12] Hu Y. and Loizou P. C. "Evaluation of objective quality measures for speech enhancement", IEEE Trans. Audio, Speech, Language Process., vol. 16, pp. 229–238. Jan 2008.