Modern Rounding Based Approximate (MROBA) Multiplier for the Design of Fir Filter

Karanam.Anusha, Ponnampalli Sudhakar Rao

Abstract— For better power utilization,area,speed which are important criteria in the design of DSP processors multiplier design is main issue.The work of multiplier is rounding the numbers both signed and unsigned.In few applications speed is more important than accuracy so to improve the performance approximate circuits are introduced with reduce energy consumption and increase speed.In this paper,we propose an FIR filter based on Modern Rounding Based Approximate(M ROBA).In this modified rounding based approximate multiplier the numbers are rounded to the nearest exponent irrespective of 2^n .The proposed MROBA multiplier based FIR filter was compared with Rounding Based Approximate (ROBA) Multiplier. In this multiplier the operands are rounded to the nearest exponent of two. This approximation will lead to simplification of multiplication operation thus reducing area and increasing speed. MROBA gives better results with the MROBA MAC unit is implemented.

Keywords: Approximate multiplier , Digital Signal Processing, Finite Impulse Response(FIR), MROBA Multiplier

I. INTRODUCTION

In the applications of DSP digital filters are used broadly to reduce noise compared to analog filters.In our the optimization of bit width and hardware resources without any impact on the frequency response and output signal precision. Addition (or subtraction), Multiplication (normally of a signal by a constant)Time Delay i.e. delaying a digital signal by one or more sample periods are three basic mathematical operations used in digital filters. The coefficients are multiplied by fixed-point constants using additions, subtractions and shifts in a multiplier block. In VLSI Signal Processing two types of digital filters are most widely used one is FIR (finite impulse response)and the other is IIR(infinite impulse response).FIR as indicates that the impulses are finite in this filter and phase is kept linear in order to noise distortions and no feedback is used for such a filters. As compared to IIR, FIR is very simple to design. Such type of FIR filters are used in DSP processors for high speed. In Digital Signal Processing Multiplication and addition is of times required. A high speed addition is done by parallel prefix adder and the better version of truncated multiplier with fewer components makes the reduction in delay. In Digital Signal Processing, FIR filters define less number of bits which are designed by using finite precision.

In IIR filter due to feedback problems will occur where as in FIR filters limited bits are efficient in which there is no feedback. Using fractional arithmetic we can implement FIR filters. The multipliers play a significant role in arithmetic operations in DSP filtering applications. The multipliers can be used in most of the DSP applications as vector product, matrix multiplication, convolution and filtering etc.In general, DSP applications requires only efficient result than accuracy. In the design we have two significant conflicts are occurring which are power consumption and propagation delay. Considering these requirements, the design of low power multiplier is of great interest. In this paper, we focus on proposing a high-speed low power/energy yet approximate multiplier appropriate for error resilient DSP applications. The contributions of this paper can be summarized as follows:

Presenting a new structure for MROBA multiplication by modifying the ROBA multiplication approach;

Describing three hardware architectures of the proposed approximate multiplication scheme for sign and unsigned operations. The proposed FIR filter consists of approximate multiplier, which is also area efficient, is constructed by modifying the conventional ROBA multiplication approximate at the algorithm level assuming rounded input values.

The main idea of ROBA approximate multiplier is to round the numbers for easy of operations as two to the power n (2n). To elaborate on the operation of the approximate multiplier, first, let the rounded numbers of the input of A and B are denoted by Ar and Br, respectively. The multiplication of A by B can be rewritten as

A*B=((Ar-A)*(Br-B))+(Ar*B)+(Br*A)-(Ar*Br) (1)

The key observation is that the multiplications of Ar × Br, A r × B, and Br × A are implemented with shift operation. The hardware implementation of (Ar − A) × (Br − B), which is complex and depends on differences of the exact numbers from their rounded ones, is typically small. Hence, we propose to omit this part from (2), helping simplify the multiplication operation.Hence, to perform the multiplication process, the following expression is used:

A*B=(Ar*B)+(Br*A)-(Ar*Br) (2)

Thus, to perform the multiplication operation using three shift and two addition/subtraction operations. In this approach, the nearest

Revised Manuscript Received on 14 August, 2019.

Karanam.Anusha, Asst. Professor, Dept. of ECE, Malla Reddy Engineering College for Women(Autonomous), Secunderabad-TS Asst.Professor,Department of ECE,RKCE
anukaranam@gmail.com
Ponnampalli.Sudhakar Rao, Asst. Professor, Dept. of ECE, Malla Reddy Engineering College for Women(Autonomous), Secunderabad-TS Asst.Professor,Department of ECE,RKCE

values for A and B in the form of 2n should be determined. When the value of A (or B) is equal to the $3 \times 2^{(p-2)}$(where p is an arbitrary positive integer larger than one), it has two nearest values in the form of 2n with equal absolute differences that are 2p and 2p−1. While both values lead to the same effect on the accuracy of the proposed multiplier, selecting the larger one (except for the case of p = 2) leads to a smaller hardware implementation for determining the nearest rounded value and hence, it is considered in this paper. It originates from the fact that the numbers in the form of $3 \times 2^{(p-2)}$ are considered as do not care in both rounding up and down simplifying the process, and smaller logic expressions may be achieved if they are used in the rounding up. It should be noted that contrary to the previous work where the approximate result is smaller than the exact result, the final result calculated by the ROBA multiplier may be either larger or smaller than the exact result depending on the magnitudes of Ar and Br compared with those of A and B, respectively. Note that if one of the operands (say A) is smaller than its corresponding rounded value while the other operand (say B) is larger than its corresponding rounded value, then the approximate result will be larger than the exact result. This is due to the fact that, in this case, the multiplication result of (Ar − A) ×(Br − B) will be negative. Since the difference between (2) and (3) is precisely this product, the approximate result becomes larger than the exact one. Finally, it should be noted the advantage of the proposed ROBA multiplier exists only for positive inputs because in the two's complement representation, the rounded values of negative inputs are not in the form of 2n..Hence, we suggest that, before the multiplication operation starts, the absolute values of both inputs and the output sign of the multiplication result based on the inputs signs be determined and then the operation be performed for unsigned numbers and, at the last stage, the proper sign be applied to the unsigned result.

II. PROPOSED MODIFIED ROUNDING BASED APPROXIMATE MULTIPLIER (MROBA)

2.1 A. Algorithm for MROBA multiplier

The main concept of the proposed modified rounding based approximate multiplier is to design the multiplier so that it takes all values which are irrespective of 2n .The detailed explanation of the multiplier is described further. Initially let us consider Ar as the rounded input input value of input A and Br as the rounded inputvalue of input B. Now the multiplication of the A* B is written has follows:

A * B= (Ar * B) (Br * B) + (Ar * B) + (Br *A) – (Ar *Br) - [1] ------ (a)

The basic key point to be considered is the product of (Ar * B) and (Br * B) is complex and the weight of the term would result in small values when compared with the exact numbers so the product of this term can omitted and it also leads to complex hardware design approach. Hence the multiplication can be performed by the following expression as

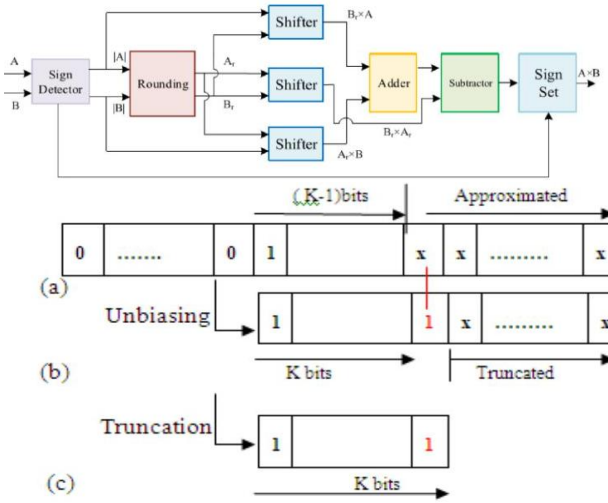


Fig:1 ROBA Multiplier Block Diagram

A * B = (Ar * B) + (Br * A) – (Ar * Br) - [1]

The product terms of (Ar * B), (Br *A), (Ar * Br) can be implemented by three barrel shifters of N bit and one adder is implemented by the parallel prefix koggestone adder of N bit and one Subtractor is also needed.

If the values of A or B is equal to N where (N=1, 2, 3….N) it has two rounded values N+1 and N. depending on the inputs of A and B the rounded values can be chosen this kind of rounding values are applicable for both A greater B (A>B) and A less than B condition (A<B). Exception that for A equal B (A=B) the rounded input of A and B would be the middle number of N+1 and N i.e. (2N+1)/2 In the conventional rounding based approximate multiplier the numbers (which represents inputs of ROBA) in the form of 3x2p-2 (where p is arbitrary positive number which is greater than one) are considered and also we have two rounded values in the form of 2p and 2p-1 both the values lead to the same effect but in this case the larger value is considered as the

rounded value for both the inputs because larger values leads in smaller hardware implementation. But the accuracy of this multiplier is poor and the exact result is not obtained in this case so,we consider the modified rounding based approximate multiplier which is applicable to all N (N=1, 2,3……N) numbers and the exact output is obtained for the given input. Fig 2.

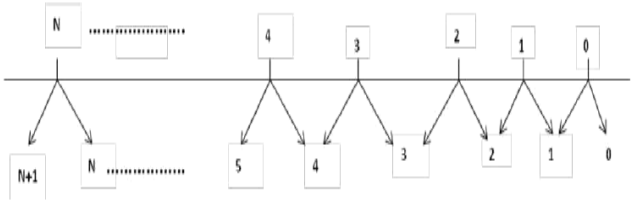


Fig 2: Top numbers represents inputs and below numbers represents the rounded values

2.2 Hardware implementation of the proposed modified rounding based approximate multiplier

Based on fig 3 we have the block diagram for the modified rounding based approximate multiplier.Initially we have a sign detector block where the inputs A and B are given. As in the case of the unsigned numbers the sign detector block can be applied to positive numbers. . But if the signed numbers are considered the absolute values are to be generated and then inputs are to be given. The input of A can have its rounded value has the larger value (N+1) or the smaller value (N). Similarly for input B the rounded value can either be the larger value or the smaller value. This condition is applicable for both A greater than B (A >B) and for A less than B (A<B) Condition. But for the A equal to B (A=B) the rounded values are considered according to (2N+1)/2.

NOTE: The rounded values of both the inputs should be compulsory differ from other rounded value eg: for input A its rounded value Ar should be larger value and for the input B the rounded value Br should be smaller value or vice versa for both inputs. Expect for (A=B) the rounded value should be the middle value or (2N+1)/2 which is same for both the inputs

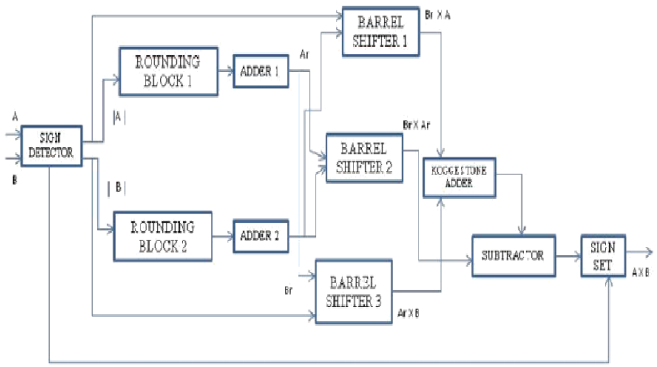


Fig3: Block diagram of the modified rounding based approximate multiplier

III. COMPARISION OF MROBA MULTIPLIER WITH ROBA MULTIPLIER COMPARED WITH THE ROBA MULTIPLIER THE MROBA GIVES THE EXACT OUTPUT FOR THE GIVEN INPUTS IN ALL THE POSSIBLE CASES.

Table 3: Comparision of MROBA multiplier with ROBA multiplier conditions ROBA MROBAmultiplier multiplier

conditions	ROBA multiplier	MROBA multiplier
A=B	Small value than the exact	Exact result
A>B	Smaller value than the exact	Exact result
A<B	Larger value than the exact	Exact result

Examples1:
ROBA multiplier: (unsigned)
When: A=12 (3* 22), B=6 (3* 21)
Ar=24 =16 Br=23 =8
A * B= (Ar* B) + (Br *A) - (Ar * Br)
Here Ar=16, Br=8
(16*6) + (8*12) – (16* 8)
72 ≠ 64

In the conventional approach the inputs should be in the 3 x 2p-1 form and we have two rounded values in the form of 2p and 2p-1 both the values lead to the same effect but here the larger value is considered as the rounded value for both the inputs because larger values
leads in smaller hardware implementation.
Examples 2:
ROBA multiplier: (signed)
When: A= -12 (3* 22), B= -12 (3*22)
Ar=24 =16 Br=24 =16
A * B= (Ar * B) + (Br * A) - (Ar * Br)
Here Ar=16, Br=16
(16*-12) + (16* -12) – (16*16)
144 ≠ 256

Here -256 is the value since we have a sign set block at the end only positive value is obtained.them with the ROBA multiplier. Most of the
parameters are considered by considering (A=B) condition
Examples 3:
MROBA multiplier: (unsigned)
1. Maximum error :
When: A= 7, B= 5
Max{error(A,B)}= ((2N+1/2)-N) *(2N+1/2)- N
Ar=7 Br=6
A * B= (Ar * B) + (Br * A) - (Ar * Br) N * N
Here Ar=7 , Br=6
(7 * 5) + (6 * 7) – (7 * 6) =1/4N2
35 = 35
2. Approximate signed error:
Examples 4:
MROBA multiplier: (signed)
Error (A, B)= (Ar-A) (Br-B) + A+B+1
Considering: A= -35, B= 47
Ar=36 Br=47
A *B = (Ar * B) + (Br * A) - (Ar * Br)
Here Ar=36, Br=47
(36 * 47) + (47 * -35) – (36 * 47)
1645 =1645
Here -1645 is the value since we have a sign set block at the end only the positive value is obtained results compared with ROBA. So the main concept here is accuracy.

So, from the above examples of MROBA multiplier and ROBA multiplier we can say that the MROBA is more accurate than ROBA and the main advantage is that we can perform multiplication for all the numbers irrespective of 2n since in the ROBA multiplier only the inputs and rounded values are strictly 2n but in this MROBA we can perform multiplication for all N inputs (N=1, 2,3……..) .

IV. FIR FILTER IMPLEMENTATION

In the FIR system, the impulse response is of finite duration, this means that it has a finite number of nonzero terms. The response of the FIR filter depends only on the present and past input samples.

A. Normal FIR filter:

The conventional design of the FIR filter is shown in Figure 8.The implementation of an FIR requires three basic building blocks: Multiplication, Addition and Signal delay. Where N represents the filter order, y [n] is the output signal and bk represents the set of filter coefficients. If x[n] is the input signal applied, x [n - k] terms are referred as taps or tapped delay lines. In the conventional design, present a simple structure of multiplier for FIR filters. It performs multiplication by generating partial products. If the multiplier digit is a 1, the multiplicand is simply copied down and represents the product. If the multiplier digit is a 0 the product is also 0. Therefore the area and delay will increase. It affects the performance of the FIR filter.

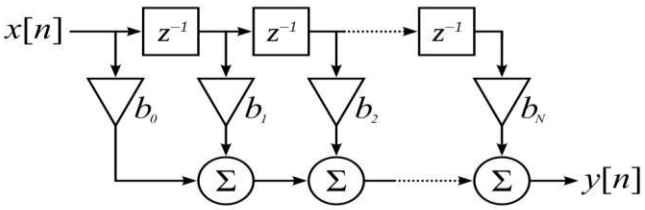


Fig 8. Block diagram of FIR filter.

V. FIR FILTER USING ROBA MULTIPLIER

As the multiplier is the slowest element in the system, it will affect the performance of the FIR filter. So, ROBA multiplier is suggested since it reduces area and it is faster than other conventional multipliers.

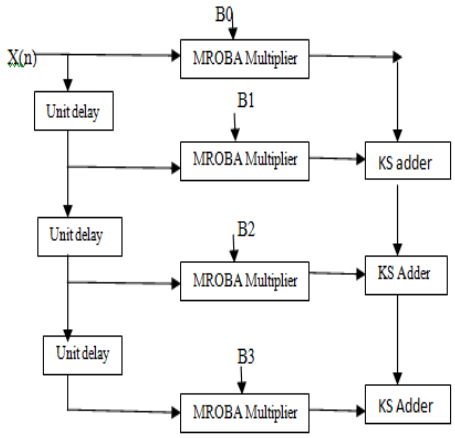
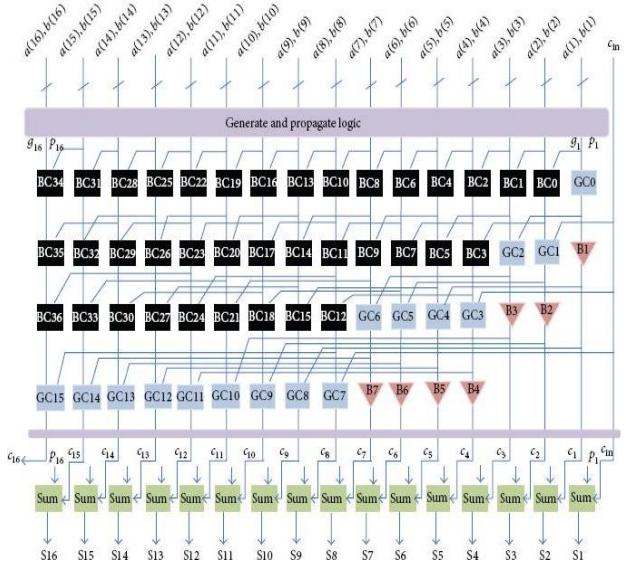


Fig. Block diagram of FIR filter using ROBA Multiplier.

Kogge-Stone adder

Kogge-Stone adder is a parallel-prefix form carry look ahead adder. Kogge-Stone adder was developed by [3] Peter M. Kogge and Harold S. Stone which they published in1973. KS adder is a fast adder design as it generate carry signal in O(log2 n) time and has the best performance in VLSI implementations. KS adder has large area with minimum fan-out which increases its performance. Kogge-Stone adder is widely used in high performance 32-bit, 64-bit, and 128-bit adders as it reduces the critical path to great extent. In fig.3 each vertical stage produce propagate and generate bits. Generate bits are produced in the last stage and XORed with initial propagate and generate bits to produce sum.



VI. CONCLUSION

An approximate multiplier(MROBA) based FIR filter implementation was proposed in the paper.The results may show that in MROBA multiplier may gives better performance in area,power and delay.

REFERENCES

1. South Asian Journal of Engineering and Technology Vol.2, No.21 (2016) 52–56
2. Nehru, K., & Linju, T. T. (2017). Design of 16 Bit Vedic Multiplier Using Semi-Custom and Full Custom Approach. Journal of Engineering Science & Technology Review, 10(2).
3. J.Surendiran, K.Subramanyam chari, T.Puvirajan," AN EXTRAORDINARILY PERCEPTIVE ULTRA-LOW-POWER APPROACH FOR LOGIC DEVICES OPERATING IN SUBTHRESHOLD VOLTAGES", IJPT| Sep-2017| Vol. 9 | Issue No.3 | 30827-30833.
4. Nehru, K., Deepthi, C., Sushma, S., & Saravanan, S. (2017). PERFORMANCE ANALYSIS OF POWER GATING TECHNIQUES IN 4-BIT SISO SHIFT REGISTER CIRCUITS. Journal of Engineering Science and Technology, 12(12), 3203-3214.
5. International Journal of Pure and Applied Mathematics Volume 118 No. 18 2018, 1539-1545
6. ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version) url: http://www.ijpam.eu
7. www.iosrjournals.org Issue 1 (September 2012), PP.031-038 www.irjes.com