IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

HIGHLY THROUGHPUT APPROXIMATE MAC BASED ON MULTIMODE OPERATIONS

[1]Mr. GOKUL M. NARAYANAN, [2]Ms. AKSA DAVID
[1]Student, [2]Assistant Professor
[1]Department of Electronics and Communication Engineering
[1]IES College of Engineering
Thrissur-Kerala, India

***Abstract*:** In various signal processing algorithms such as machine learning and multimedia digital signal processing, energy consumption can be minimized by the use of approximate computing, since these applications are generally known to have error tolerable characteristics. A novel approximate computing scheme suitable for realizing a double throughput energy-efficient multiply-accumulate (MAC) processing is presented in this paper. In this work, different approximate multipliers are used in an interleaved way to compensate errors in the opposite direction during accumulate operations, which minimizes the error accumulation limiting the approximate range present in the previous works. For the balanced error accumulation, the approximate 4-2 compressors generating errors in the opposite direction are designed and based on the probabilistic analysis, positive and negative multipliers are then carefully developed to provide a similar error distance. The new architecture is extended to create a versatile double throughput MAC (DTMAC) unit that efficiently performs either Multiply-Accumulate or simple Multiplication operation. The new architecture is further extended to create a versatile double-throughput MAC (DTMAC) unit that efficiently performs either multiply-accumulate or multiply operations for N-bit, 1xN/2 - bit, or 2xN/2-bit operands. In comparison to a fixed-function 16-bit MAC unit, 8-bit multiply- accumulate operations can be executed with higher energy efficiency on a 16-bit DTMAC unit.

***Index Terms*: Approximate computing, double throughput, energy efficient, variable word length**

I. INTRODUCTION

Approximate computing is an emerging paradigm for energy-efficient and/or high performance design. It includes a plethora of computation techniques that return a possibly inaccurate result rather than a guaranteed accurate result, and that can be used for applications where an approximate result is sufficient for its purpose. Approximate computing is based on the observation that in many scenarios, although performing exact computation requires large amount of resources, allowing bounded approximation can provide disproportionate gains in performance and energy, while still achieving acceptable result accuracy.

Basically approximate adders, multipliers, MAC units etc., are used in signal processing (FIR Filters, IIR Filters, etc.) and image processing applications (Image Sharpening, Image Smoothening, etc.). In all these applications, the fundamental blocks are adders and multipliers. Normally a large number of adders and multipliers have to be used according to the circuit requirement. So, if we go for exact adders and multipliers in the above applications, power consumption, area and circuit complexity will be very high. In order to overcome this issue, we go for approximation. In approximate adders and multipliers, we divide the circuit design into two parts - Least Significant Bit (LSB) part and Most Significant Bit (MSB) part. The MSB part is kept unchanged or exact and the approximation is performed in the LSB part. This is because if we try to approximate the MSB part, the resulting error will be so high.

The main advantages of approximate circuits includes less area requirement, delay can be minimized and the overall power requirement will be less compared to exact circuits. The only problem with the approximate design is that since we are approximating the exact logic of the system, errors might occur at the final output. But, for all the above mentioned applications, very minimum errors will be occurring in the LSB part and can be negligible. These errors in LSB part will not affect the overall system function.

II. RELATED WORK

The interleaving scheme using positive and negative compressors alternatively as explained in [1] with opposite error directions can generate a balanced error distribution during the MAC operation. This method efficiently offsets the accumulated errors due to the opposite error directions of compressors used.

The approximate multiplier design explained in [4] is based on some equality that transforms the sum of two partial products in to some coded form which can be implemented using only AND - OR gates. Thus, use of XOR gates can be limited to a large extend which in turn speeds up the whole circuit performance since XOR gates has larger delay compared to AND - OR gates.

The general construction of MAC unit is explained in [5]. Among the three stages of MAC unit (partial product generator, summation tree and final adder), the summation network acts as the core of MAC unit and this part consumes most of the circuit area and power and causes the overall delay in the circuit. Overall optimization of whole circuit can be easily achieved by optimizing the summation network.

Compressors are XOR rich circuit and partial product reduction is the critical multiplier block in terms of speed, area and power. The approximate compressor design in [6] uses only AND - OR gates and can be used to replace some of the exact compressors, with minimum error probability and average error. The electrical performance and the arithmetic accuracy of the binary multiplier depend on the allocation of the approximate compressors.

Word length optimization method discussed in [7] points to the need for using fewest number of bits possible to carry each signal in a digital signal processing system. One has to be more careful while using the word length optimization technique since the quantization noise may degrade the whole system performance if the number of bits used is too small.

Using a variable word length as explained in [8] can be an effective approach for optimizing the word length to match the conditions. The twin precision technique in [3] is basically a variable word length method which helps us to use a single circuit in multiple modes of operations. In twin precision technique, any size of two smaller operations can be chosen as long as the precision of the two smaller operations together are equal or smaller than the full precision of the designed circuit.

III. TWIN PRECISION MULTIPLIERS

Multiplication is a complex arithmetic operation, which possesses relatively high signal propagation delay, higher power dissipation and large area requirement. While choosing a multiplier for a digital application, the bit width of the multiplier is required to be at least as wide as the largest operand of the applications that has to run on the digital system. The bit width of the multiplier is, therefore, often much larger than its operands which leads to excessive power dissipation and delay. Multipliers can be classified based on which data words are accessed, namely, serial form, parallel form, serial – parallel form. In high performance digital systems such as microprocessors, FIR filters and digital signal processors etc., the multiplier is one of the key hardware blocks. So the design of multipliers stands challenging with advancement in technology.

Twin Precision multiplier is a type of unsigned multiplication process. It can be implemented using both unsigned and signed multiplication process. In an unsigned binary multiplication each bit of one of the operands called the multiplier is multiplied with the second operand called the multiplicand. Thus one row of partial products is generated. Each row of partial product is shifted according to the position of the bit of the multiplier forming what is commonly called the partial product array. Finally, partial products that are in the same column are summed together, forming the final result. But the unsigned multiplication is of limited use. It can be implemented using signed multiplication process through algorithm called Baugh-Wooley[2] (BW).

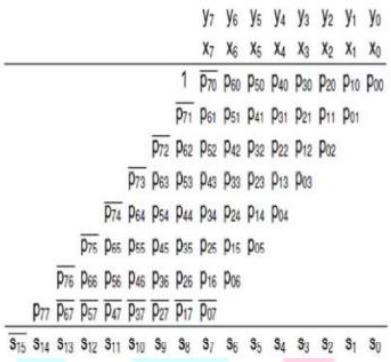

Fig 1. Block Diagram of Twin Precision Multiplier using Baugh Wooley

The BW algorithm is a direct way of doing signed multiplications. The creation of the reorganized partial product array comprises of three steps:

i) The most significant partial product of the first N- 1 rows and the last row of partial products except the most significant has to be negated,

ii) A constant one is added to the Nth column,

iii) The most significant bit (MSB) of the final result is negated.

The twin-precision technique is an efficient way of achieving double throughput in a multiplier with low area overhead and with only a small delay penalty[3]. In twin precision technique, any size of the two smaller multiplications can be chosen, as long as the precision of the two smaller multiplications together are equal or smaller than the full precision of the multiplication.

IV. PROPOSED TECHNOLOGY

A highly throughput approximate MAC based on multimode operations is being proposed in this work. The proposed circuit work as a multi precision network allowing us to select the operational bit length as per our requirement.

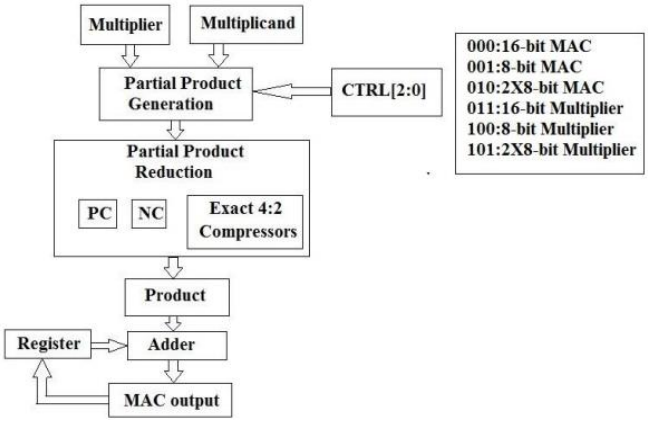

Fig 2. Block Diagram of Proposed Multi Precision MAC

Figure 2 shows the basic block diagram of the proposed multi precision double throughput Multiply – Accumulate unit. A 16x16 MAC has been designed as two 8x8 MAC units and according to the requirement of the circuit parameters, the MAC units are activated. That is, for some applications, if only a 8x8 MAC operation is sufficient, then from the two 8x8 MAC units, one can be made active while keeping the other one idle. If the circuit requires both the 8 bit MAC units, then both can be activated simultaneously, providing a double throughput from a single circuit arrangement.

TABLE 1. Operating Modes

000	Full Precision 16 bit MAC
001	Half Precision 1x8 bit MAC (Approximate)
010	Double Throughput 2x8 bit MAC
011	Full Precision 16 bit Multiplication
100	Half Precision 1x8 bit Multiplication (Approximate)
101	Double Throughput 2x8 bit Multiplication
110	Exact 8 bit Multiplier
111	Exact 8 bit MAC

Apart from the above discussed operation, the proposed circuit can be used in six different ways as per the user select/control inputs. The same circuit can be used as simple multiplier or as MAC unit itself. The table I shows the operation modes of the proposed circuit.

4.1 Interleaving Scheme

The method of using positive and negative compressors alternatively as shown above is called interleaving scheme of approximate multipliers with the opposite error directions, generating the balanced error distribution during MAC operations[1]. Without applying the identical multiplier architecture, one can simply introduce the interleaved multiplication sequences associated with PM and NM, which efficiently offset the accumulated errors by the opposite error directions.

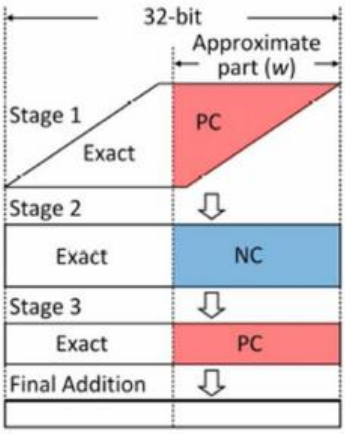

Fig 3. Architecture of a 16x16 positive multiplier using interleaving scheme[1]

4.2 Multiplu and Accumulate

In computing, especially digital signal processing, the multiply–accumulate (MAC) operation is a common step that computes the product of two numbers and adds that product to an accumulator. The hardware unit that performs the operation is known as a multiplier– accumulator (MAC unit); the operation itself is also often called a MAC or a MAC operation. The MAC operation modifies an accumulator a

$$a \leftarrow a + (b \times c)$$

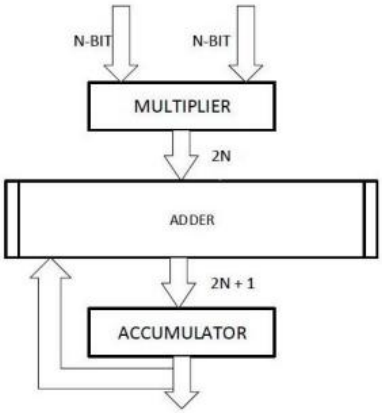

Fig 4. Basic MAC operation

Modern computers may contain a dedicated MAC, consisting of a multiplier implemented in combinational logic followed by an adder and an accumulator register that stores the result. The output of the register is fed back to one input of the adder, so that on each clock cycle, the output of the multiplier is added to the register. Combinational multipliers require a large amount of logic, but can compute a product much more quickly than the method of shifting and adding typical of earlier computers.

V. EXPERIMENTAL RESULTS

ModelSim software is used for coding and simulating the existing and proposed architectures. ISE Design suite can also be used but the ModelSim software is more user friendly and it has an inbuilt simulation environment, which most of the ISE design suite older versions failed to provide. The code has been written in Verilog Description language and simulated successfully for obtaining the output.

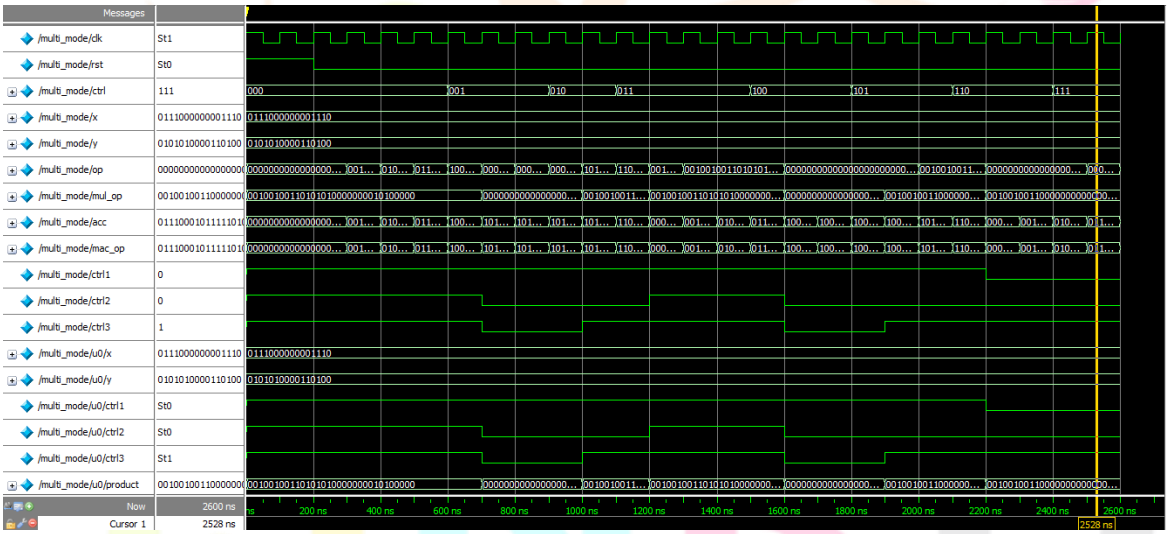

Fig 5. Simulated output of proposed multi precision MAC

VI. PERFORMANCE COMPARISON

Xilinx ISE 8.1i was used to evaluate the clock period and maximum frequency of the existing MAC architecture and the proposed multi precision MAC architecture. Table I shows the comparison between the two designs.

TABLE 2. Comparison of Clock period and maximum frequency

	Existing MAC architecture	Proposed multi precision MAC architecture
Clock	32	84
Clock Period (ns)	5.587	22.383
Maximum Frequency (MHz)	178.987	44.677

VII. CONCLUSION

A novel approximate computing scheme suitable for realizing a double throughput energy-efficient multiply-accumulate (MAC) processing has been designed and simulated successfully using ModelSim software. In this work, different approximate multipliers are used in an interleaved way to compensate errors in the opposite direction during accumulate operations, which minimizes the error accumulation limiting the approximate range present in the previous works. For the balanced error accumulation, the approximate 4-2 compressors generating errors in the opposite direction are designed and based on the probabilistic analysis, positive and negative multipliers are then carefully developed to provide a similar error distance. The proposed architecture is extended to create a versatile multi precision double throughput MAC (DTMAC) unit that efficiently performs either Multiply-Accumulate or simple Multiplication operation.

REFERENCES

[1] Gunho Park, Jaeha Kung and Youngjoo Lee, "Design and Analysis of Approximate Compressors for Balanced Error Accumulation in MAC Operator", IEEE Transactions On Circuits And Systems—I, 2021

[2] Spoorthi H R, Dr. Narendra C P, Chandra Mohan U, "Low Power Datapath Architecture for Multiply – Accumulate (MAC) Unit", 2019 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT-2019), MAY 17th & 18th 2019

[3] Magnus Själander and Per Larsson-Edefors, "Multiplication Acceleration Through Twin Precision", IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol. 17, No. 9, September 2009

[4] Manzhen Wang, Yuanyong Luo, Mengyu An, Yuou Qiu, Muhan Zheng, Zhongfeng Wang and Hongbing Pan, "An Optimized Compression Strategy for Compressor-based Approximate Multiplier", IEEE 2020

[5] A. Abdelgawad, Magdy Bayoumi, "High Speed and Area-Efficient Multiply Accumulate (MAC) Unit for Digital Signal Processing Applications", IEEE 2007

[6] Darjn Esposito , Antonio Giuseppe Maria Strollo , Ettore Napoli , Davide De Caro, and Nicola Petra, "Approximate Multipliers Based on New Approximate Compressors", IEEE Transactions on Circuits and Systems–i: Regular Papers, 2018

[7] Wonyong Sung and Ki-I1 Kum, "Simulation-Based Word-Length Optimization Method for Fixed-point Digital Signal Processing Systems", IEEE Transactions On Signal Processing, Vol. 43, No. 12, December 1995

[8] Shingo Yoshizawa and Yoshikazu Miyanaga, "Use of a Variable Word length Technique in an OFDM Receiver to Reduce Energy Dissipation", IEEE 2007.