Extensive Review on Pre-Encoded Multipliers Based on Non-Redundant Radix-4 Signed-Digit Encoding

Govind Singh Solanki[1], Prof. Devesh Kishore[2], Prof. Aastha Hajari[3]

[1]Mtech. Research Scholar, [2]Research Guide, [3]HOD Department of Electronics and Communication

SKSITS Indore

Abstract - Multipliers are one of the most important elements of many systems with high performance such as microprocessors, FIR filter, DSP, etc. Multiplier is considered as the slowest element thus it determines the overall performance of the system. Above all, it consumes high area. Hence, major design issue is to obtain optimization between the multiplier's speed and area. Larger area is an effect of improvement in the speed, therefore making area and speed a conflicting constraints. This research presents an extensive survey on pre encoded multipliers based on radix-4 signed encoding. With the increasing level of complexities and device integration of microelectronic circuits, power dissipation delay and area reduction has also come up as a primary design goal.

Keywords- Digital multipliers, Pre-Encoded Multipliers, Signed-Digit Encoding, Non-Redundant Radix-4.

I. INTRODUCTION

Today multiplier is using in every basic circuit. All the ALU system is based on multipliers. Complete arithmetic Logical part is based on a multiplier and if multiplier is consuming so much delay then the entire product which is based on a multiplier is fail due to fail of multiplier.

If multiplier have low speed then it will works slowly. Regarding this if our function is working in 2 second then it will also take some delay .Then its output will be 2second+ delay. That delay may greater then to basic performing delay.

In VLSI speed of any IC is depend on power consumption, Area, delay. Some of them have a complex circuit and at that time will get increment in delay and power consumption. Power consumption is also a main power factor If reduce the power factor of an IC the it is showing that our product battery life is good.

Today everybody is using calculators and CPU. Every company is working for low power consumption circuit so that they can deliver more long life battery as comparison to other company.

If our Multiplier power consumption will be increase then heat dissipation will be increase. So it will increase leakage current. So this multiplier will be used in many ALU circuits. then all the product of this ALU have a low battery life .If multiplier operations are used for memory allocation of mobile phone then battery of that mobile phone will not be long life because the heat dissipation, power consumption is greater than the normal range .

Many DSP applications demand high throughput and real-time response, performance constraints that often dictate unique architectures with high levels of concurrency. DSP designers need the capability to manipulate and evaluate complex algorithms to extract the necessary level of concurrency. Performance constraints can also be addressed by applying alternative technologies. A change at the implementation level of design by the insertion of a new technology can often make viable an existing marginal algorithm or architecture.

For implementing a digital multiplier a large variety of computer arithmetic algorithms could be used. Most techniques take into consideration generating a set of partial products, and then adding the partial products together once they have been shifted. In a multiplier to increase its speed, the number of partial product to be generated should be reduced. A higher representation radix effectively indicates to fewer digits. Thus, a single digit multiplication algorithm necessitates fewer cycles as moving to much higher radices, which automatically leads to a lesser number of partial products. Several algorithms have been developed for this purpose like Booth's Algorithm, Wallace Tree method etc. For the summation process several adder architectures are available viz. Ripple Carry Addition, Carry Look-ahead Addition, Carry Save Addition etc. But to reduce the power consumption the summation architecture of the multiplier should be carefully chosen.

According to Moore low in every 18 month the transistors of any IC will be doubled which is using in a IC. According to moore low after every 18 month there will be a new IC in which will find a ore number of transistor according to previous .Now there is a question that what is benefit from this increase in number of transistor. If

number of transistor will increase that means you have to increase its functionality. Right now our IC can work more and fast work as comparison to previous IC refer to figure 1.1 moors low plot.

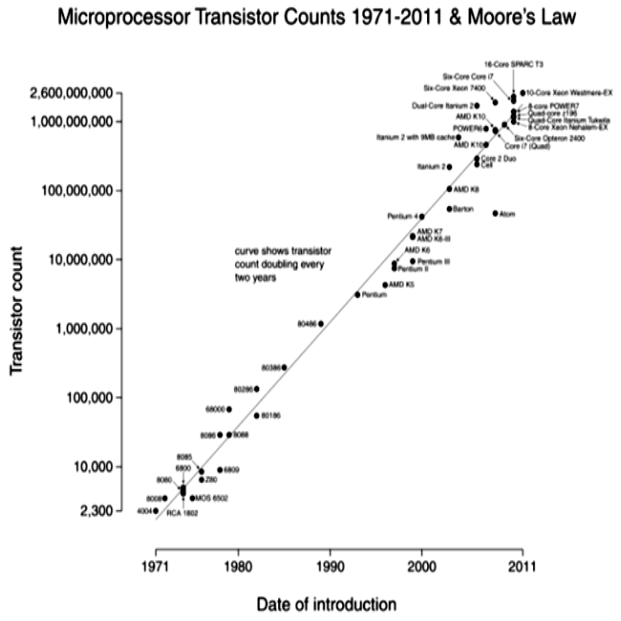

Figure 1.1 Date of introduction of technology.

II. SYSTEM MODEL

Though Wallace Tree multipliers were faster than the traditional Carry Save Method, it also was very irregular and hence was complicated while drawing the Layouts. Slowly when multiplier bits gets beyond 32-bits large numbers of logic gates are required and hence also more interconnecting wires which makes chip design large and slows down operating speed

Booth multiplier can be used in different modes such as radix-2, radix-4, radix-8 etc. To use Radix-4 Booth's Algorithm because of number of Partial products is reduced to n/2.

One of the solutions realizing high speed multipliers is to enhance parallelism which helps in decreasing the number of subsequent calculation stages. The Original version of Booth's multiplier (Radix – 2) had two drawbacks.

The number of add / subtract operations became variable and hence became inconvenient while designing Parallel multipliers.

A. Types of Multipliers

There are a multiple type of multipliers discussed in this section. The most general multiplication method is based on the "add and shift" algorithm. The various types of multipliers are mentioned in the following points-

Serial multiplier, Parallel multiplier, Shift and Add multiplier, Combinational multiplier, Wallace multiplier, Modified Wallace multiplier, Array multiplier, Booth multiplier, Modified Booth multiplier, Sequential Decimal multiplier.

B. Basic Stages of Multiplication

There are three basic steps to perform multiplication operation are listed below.

1) *Partial Product Generation*
2) *Partial Product Accumulation*
3) *Carry-propagate addition*

Decimal multiplication is considered as the most complex operation. Its complexity level is mainly higher than the binary multiplication because of the following two reasons: the decimal digits have higher range, which increases the number of multiples in the multiplicand, thereby decreases the efficiency of decimal values represented in BCD–8421 coding mechanism.

1. Partial Product Generation

The partial products are generated easily by making use of the generated easy multiples i.e. 2X, 4X, 5X. Easy multiples are generated so as to ignore the operation on negative numbers.

2. Partial Product Accumulation

The direct approach of implementing the PPA is to join two of the partial products at the same time, add them using an adder and repeat the process until the final result is attained. If a single adder is used to perform all these operations, then this reduction of partial product will consume only N- cycles which mean a total consumption of N- mechanical delays if the relay adder is used.

3. Carry Propagate Addition

The basic usage of a carry save adder is to accumulate the partial products resulting in the final sum and carry. It improves the speed of accumulation of the partial product as it saves the carry and passes it to the next level of carry select adder. Therefore, the adders in the same layer become independent of each other and can be executed simultaneously. Hence the time required for the addition operation is reduced.

The carry save adder tree uses a one's compliment based radix-2 modified booth algorithm for partial product generation and accumulation. Accumulation is combined with a carry save adder tree to compress the partial products.

III. LITERATURE REVIEW

SR. NO	TITLE	AUTHOR	YEAR	APPROACH
1	Pre-Encoded Multipliers Based on Non-Redundant Radix-4 Signed-Digit Encoding,"	K. Tsoumanis, N. Axelos, N. Moschopoulos, G. Zervakis and K. Pekmestzi,	2016	introduce an architecture of pre-encoded multipliers based on off-line encoding of coefficients
2	Performance analysis of Wallace and radix-4 Booth-Wallace multipliers,	S. Asif and Yinan Kong,	2015	The radix-4 Booth-Wallace and the Wallace multipliers are implemented for various sizes.
3	An Efficient Softcore Multiplier Architecture for Xilinx FPGAs,	M. Kumm, S. Abbas and P. Zipf,	2015	Efficient implementation of a softcore multiplier, i. e., a multiplier architecture which can be efficiently mapped to the slice
4	A New Redundant Binary Partial Product Generator for Fast 2n-Bit Multiplier Design,	C. Xiaoping, H. Wei, C. Xin and W. Shumin,	2014	A new radix-16 RB Booth Encoding (RBBE-4) to avoid the hard multiple of high-radix Booth encoding without incurring any ECW
5	Radix-4 and radix-8 booth encoded interleaved modular multipliers over general Fp,	K. Javeed and Xiaojun Wang,	2014	Presents radix-4 and radix-8 Booth encoded modular multipliers
6	Radix-4 and Radix-8 Booth Encoded Multi-Modulus Multipliers,"	R. Muralidharan and C. H. Chang,	2013	Employs Booth encoded modulo and modulo multiplier architectures.
7	A Design and Implementation of Decimal Floating-point Multiplication Unit Based on SOPC,	H. Ding, P. Shu, X. Wang and J. Yang,	2012	Signed-Digit radix-4 algorithm and new BCD coding techniques for the decomposition of decimal floating- point computing

K. Tsoumanis, N. Axelos, N. Moschopoulos, G. Zervakis and K. Pekmestzi,[1] In this paper, introduce an architecture of pre-encoded multipliers for Digital Signal Processing applications based on off-line encoding of coefficients. To this extend, the Non-Redundant radix-4 Signed-Digit (NR4SD) encoding technique, which uses the digit values {−1, 0, +1, +2} or {−2, −1, 0, +1}, is proposed leading to a multiplier design with less complex partial products implementation. Extensive experimental analysis verifies that the proposed pre-encoded NR4SD multipliers, including the coefficients memory, are more area and power efficient than the conventional Modified Booth scheme.

S. Asif and Yinan Kong, [2] Multiplication is one of the most commonly used operations in the arithmetic. Multipliers based on Wallace reduction tree provide an area-efficient strategy for high speed multiplication. In the previous years the Booth encoding is widely used in the tree multipliers to increase the speed of the multiplier. However, the efficiency of the Booth encoders decreases with the technology scale down. In this research work showed that the use of Booth encoders in fact increases the delay and power of the Wallace multiplier in the deep submicron technology. The radix-4 Booth-Wallace and the Wallace multipliers are implemented for various sizes and synthesized using Synopsys Design Compiler in 90nm process technology. The synthesis results show that the Wallace multiplier has up to 17% less delay and 70% less power consumption as compared to the radix-4 Booth-Wallace multipliers. The Power-Delay Product (PDP) of the Wallace multiplier is up to 68% lower than the Booth-Wallace multiplier.

M. Kumm, S. Abbas and P. Zipf [3] This work presents an efficient implementation of a softcore multiplier, i. e., a multiplier architecture which can be efficiently mapped to the slice resources of modern Xilinx FPGAs. Instead of dividing the multiplication into the generation of partial products and the summation using a compressor tree, as done in modern multipliers, an array-like architecture is proposed. Each row of the array generates a partial product which is directly added to results of previous rows using the fast carry chain. A radix-4 Booth encoding/decoding is used to reduce the I/O count of the partial product generation which makes it possible to map both, the Booth encoder and decoder, into a single 6-input look up table (LUT). Like a conventional Booth multiplier, this nearly halves the number of rows compared to a ripple carry array multiplier. In addition, the compressor tree is completely avoided and an efficient and regular structure retains that uses up to 50 % less slice resources compared

to previous approaches and offers a multiply accumulate (MAC) operation without extra resources..

C. Xiaoping, H. Wei, C. Xin and W. Shumin [4] The radix-4 Booth encoding or Modified Booth encoding (MBE) has been widely adopted in partial products generator to design high-speed redundant binary (RB) multipliers. Due to the existence of an error-correcting word (ECW) generated by MBE and RB encoding, the RB multiplier generates an additional RB partial product rows. An extra RB partial product accumulator (RBPPA) stage is needed for 2n-b RB MBE multiplier. The higher radix Booth algorithm than radix-4 can be adopted to reduce the number of partial products. However, the Booth encoding is not efficient because of the difficulty in generating hard multiples. The hard multiples problem in RB multiplier can be resolved by difference of two simple power-of-two multiples. This research work presents a new radix-16 RB Booth Encoding (RBBE-4) to avoid the hard multiple of high-radix Booth encoding without incurring any ECW. The proposed method leads to make high- The proposed method leads to make high show that the proposed RBBE-4 multiplier achieves significant improvement in delay and power consumption compared with the RB MBE multiplier and the current reported best RBBE-4 multipliers.

K. Javeed and Xiaojun Wang,[5] This research work presents radix-4 and radix-8 Booth encoded modular multipliers over general Fp based on inter- leaved multiplication algorithm. An existing bit serial interleaved multiplication algorithm is modified using radix-4, radix-8 and Booth recoding techniques. The modified radix-4 and radix-8 versions of interleaved multiplication result in 50% and 75% reduction in required number of clock cycles for one modular multiplication over the corresponding bit serial interleaved multipliers, while maintaining a competitive critical path delay. The proposed architectures are implemented in Verilog HDL and synthesized by targeting virtex-6 FPGA platform. Due to an efficient utilization of optimized addition chains available in FPGAs and exploiting the parallelism among operations, the proposed radix-4 and radix-8 multipliers compute one 256 256 bit modular multiplication in 1.49µs and 0.93µs respectively, which are 35% and 94% improvement over the corresponding bit serial version. Further, this work also presents a thorough comparison on basis of area, throughput, and area time per bit value. Which shows that these designs are efficiently optimized for area time per bit value with a high throughput rate. Thus, these designs are suitable to construct most of the elliptic curve and pairing based cryptographic processors.

R. Muralidharan and C. H. Chang,[6] Novel multi-modulus designs capable of performing the desired modulo operation for more than one modulus in Residue Number System (RNS) are explored in this research work to lower the hardware overhead of residue multiplication. Two multi-modulus multipliers that reuse the hardware resources amongst the modulo, modulo and modulo multipliers by virtue of their analogous number theoretic properties are proposed. The former employs the radix-Booth encoding algorithm and the latter employs the radix-Booth encoding algorithm. In the proposed and radix-Booth encoded multi-modulus multipliers, the modulo-reduced products for the moduli, and are computed successively. With the basis of the radix- Booth encoded modulo and radix- Booth encoded modulo and modulo multiplier architectures, new Booth encoded modulo multi-pliers are proposed to maximally share the hardware resources in the multi-modulus architectures. Our experimental results on based RNS multiplication show that the proposed radix- and Booth encoded multi-modulus multipliers save nearly 60% of area over the corresponding single-modulus multipliers. The proposed and Booth encoded multi-modulus multipliers increase the delay of the corresponding single-modulus multipliers by 18% and 13%, respectively in the worst case. Compared to the single-modulus multipliers, the proposed multi-modulus multipliers incur a minor power dissipation penalty of 5%.

H. Ding, P. Shu, X. Wang and J. Yang,[7] Processor design is a widely studied topic in computer system architecture design. How to improve computer performance is an important part of the computer overall design. In general processors, multiplication components play a decisive role in processor's performance. An important and frequent operation in decimal computations is multiplication. However, due to the inherent inefficiency of decimal arithmetic implementations in binary logic, practically all the proposed decimal multipliers are sequential units. Binary computing couldn't be avoided of conversion efficiency lowly and loss of accuracy. In this research work direct expanding the decimal computing applications and binary can't meet the needs of decimal operations, according to this new standard IEEE-754r, use SOPC technology design and implement a new architecture based on the decimal floating-point multiplication unit. This design takes advantage of flexibility and low-power of SOPC, the independence of IP core and so on; it is packaged as an independent IP core. This decimal floating-point multiplication unit is broadly applications in the general processors, portable devices, and mass data processing and so on. It uses Signed-Digit radix-4 algorithm and new BCD coding techniques for the decomposition of decimal floating- point computing. and compared with the common single- precision binary floating-point unit, it was wider computing, higher accuracy, faster computing speed and wider application.

The main contributions of this research work include: (1) Customized a 32/64 bit fully functional decimal floating point multiplication IP core; (2) Improved partial products based on the BCD-8421 and revised parts of the circuit; (3) According to the customized component operational requirements, defined a way of data bus, caused decimal floating point multiplication unit is good access SOPC system bus. This unit can be well used to processors, which support the standard of decimal floating-point operations, to improve processor performance. This model is verified by synthesis to Altera's low cost Cyclone Ċ FPGA.

IV. PROBLEM FORMULATION

Multimedia and Digital Signal Processing (DSP) applications (e.g., Fast Fourier Transform (FFT), audio/video CoDecs) carry out a large number of multiplications with coefficients that do not change during the execution of the application. Since the multiplier is a basic component for implementing computationally intensive applications, its architecture seriously affects their performance. The size of ROM used to store the groups of coefficients is significantly reduced as well as the area and power consumption of the circuit. However, this multiplier design lacks flexibility since the partial products generation unit is designed specifically for a group of coefficients and cannot be reused for another group. Also, this method cannot be easily extended to large groups of pre-determined coefficients attaining at the same time high efficiency [1]. Pre-encoded NR4SD multipliers, including the coefficients memory, are more area and power efficient than the conventional Modified Booth scheme.

V. CONCLUSION

This work presents an extensive survey of pre encoded multipliers. Over the past few decades, a number of researchers used Booth encoders to reduce the delay of the tree multipliers due to their smaller size of partial product tree. Some of the Booth encoder based high speed multipliers are reviewed and analyzed there performance based on result available in literature. The work was motivated by the ever increasing use of Booth encoding in the literature to reduce the delay of the multiplier. Also analyzed the area occupied and the time delay consumed by different adders and found out an appropriate relationship among the time and area complexity the adders which have taken into consideration based on their literature survey.

REFERENCES

[1] K. Tsoumanis, N. Axelos, N. Moschopoulos, G. Zervakis and K. Pekmestzi, "Pre-Encoded Multipliers Based on Non-Redundant Radix-4 Signed-Digit Encoding," in IEEE Transactions on Computers, vol. 65, no. 2, pp. 670-676, Feb. 1 2016.

[2] S. Asif and Yinan Kong, "Performance analysis of Wallace and radix-4 Booth-Wallace multipliers," 2015 Electronic System Level Synthesis Conference (ESLsyn), San Francisco, CA, 2015, pp. 17-22.

[3] M. Kumm, S. Abbas and P. Zipf, "An Efficient Softcore Multiplier Architecture for Xilinx FPGAs," 2015 IEEE 22nd Symposium on Computer Arithmetic, Lyon, 2015, pp. 18-25.

[4] C. Xiaoping, H. Wei, C. Xin and W. Shumin, "A New Redundant Binary Partial Product Generator for Fast 2n-Bit Multiplier Design," 2014 IEEE 17th International Conference on Computational Science and Engineering, Chengdu, 2014, pp. 840-844.

[5] K. Javeed and Xiaojun Wang, "Radix-4 and radix-8 booth encoded interleaved modular multipliers over general Fp," 2014 24th International Conference on Field Programmable Logic and Applications (FPL), Munich, 2014, pp. 1-6.

[6] R. Muralidharan and C. H. Chang, "Radix-4 and Radix-8 Booth Encoded Multi-Modulus Multipliers," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 60, no. 11, pp. 2940-2952, Nov. 2013.

[7] H. Ding, P. Shu, X. Wang and J. Yang, "A Design and Implementation of Decimal Floating-point Multiplication Unit Based on SOPC," 2012 Third International Conference on Digital Manufacturing & Automation, GuiLin, 2012, pp. 36-41.

[8] C. Wang, W.-S. Gan, C. C. Jong, and J. Luo, "A low-cost 256-point fft processor for portable speech and audio applications," in Int. Symp. on Integrated Circuits (ISIC 2007), Sep. 2007, pp. 81–84.

[9] A. Jacobson, D. Truong, and B. Baas, "The design of a recon- figurable continuous-flow mixed-radix fft processor," in IEEE Int. Symp. on Circuits and Syst. (ISCAS 2009), May 2009, pp. 1133–1136.

[10] Y. T. Han, J. S. Koh, and S. H. Kwon, "Synthesis filter for mpeg-2 audio decoder," Patent US 5 812 979, Sep., 1998.

[11] M. Kolluru, "Audio decoder core constants rom optimization," Patent US 6 108 633, Aug., 2000.

[12] H.-Y. Lin, Y.-C. Chao, C.-H. Chen, B.-D. Liu, and J.-F. Yang, "Combined 2-d transform and quantization architectures for h.264 video coders," in IEEE Int. Symp. on Circuits and Syst. (ISCAS 2005), vol. 2, May 2005, pp. 1802–1805.

[13] G. Pastuszak, "A high-performance architecture of the double- mode binary coder for h.264.avc," IEEE Trans. Circuits Syst. Video Technol., vol. 18, no. 7, pp. 949–960, Jul. 2008.

[14] J. Park, K. Muhammad, and K. Roy, "High-performance fir filter design based on sharing multiplication," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 11, no. 2, pp. 244–253, Apr. 2003.