International Journal of Science Engineering and Advance Technology

Implementation of High Speed Area Efficient Fixed Width Multiplier

G.Rakesh, R. Durga Gopal , D.N Rao
MTECH(VLSI), JBREC Associate Professor , JBREC Principal
rakhesh.golla@gmail.com , rdurgagopal@gmail.com , principal_jbr@yahoo.com
Joginapally BR Engineering College,Yenkapally, Moinabad Mandal , R.R. District, Hyderabad

Abstract- **The aim of project is to design a proposed truncated multiplier with less area utilization and low power comparing with previous multipliers. The proposed method finally reduces the number of full adders and half adders during the tree reduction. While using this proposed method experimentally, area can be saved. The output is in the form of LSB and MSB. Finally the LSB part is compressed by using operations such as deletion, reduction, truncation, rounding and final addition. In previous system, to reduce the truncation error by adding error compensation circuits. In this project truncation error is not more than 1 ulp (unit of least position). So there is no need of error compensation circuits, and the final output will be précised.**

Key Words—Computer arithmetic, faithful rounding, fixed- width multiplier, tree reduction, truncated multiplier.

I. INTRODUCTION

Multipliers play an important role in today's Digital Signal Processing (DSP) and various other applications. With advances in technology, many researchers have tried and are trying to design multipliers which offer either of the following design targets – high speed, low power consumption, regularity of layout and hence less area or even combination of them in one multiplier thus making them suitable for various high speed, low power and compact VLSI implementation.

The common multiplication method is "add and shift" algorithm. In parallel multipliers number of partial products to be added is the main parameter that determines the performance of the multiplier. To reduce the number of partial products to be added, Modified Booth algorithm is one of the most popular algorithms. To achieve speed improvements Wallace Tree algorithm can be used to reduce the number of sequential adding stages.On the other hand "serial-parallel" multipliers compromise speed to achieve better performance forarea and power consumption. The selection of a parallel or serial multiplier actually depends on the nature of application.

In previous method, multiplication of two bits produces an output which is twice that of the original bit. It is usually needed to truncate the partial product bits to the required precision to reduce area cost. Fixed-width multipliers, a subset of truncated multipliers, compute only n most significant bits (MSBs) of the 2n-bit product for n × n multiplication and use extra correction/compensation circuits to reduce truncation errors.To reduce the truncation error by adding error compensation circuits. So that the output will be précised.

In Proposed method jointly considers the tree reduction, truncation, and rounding of the PP bits during the design of fast parallel truncated multipliers so that the final truncated product satisfies the precision requirement.In this method truncation error is not more than 1ulp (unit of least position), so there is no need of error compensation circuits, and the final output will be précised.

II. BLOCK DIAGRAM OF BASIC MULTIPLIER

Basic Multiplier consists of various blocks such as

1. Partial Product Generation
2. Partial Product Reduction
3. Carry Propagate Addition (CPA)

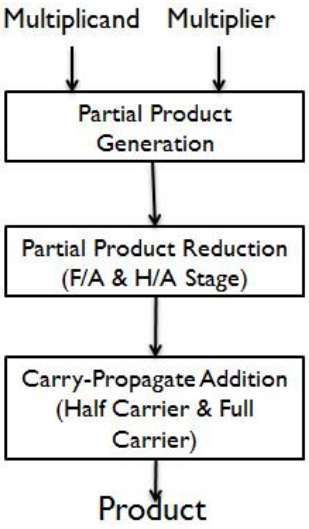


Fig.1 Block Diagram of Basic Multiplier

PP (partial product) generation produces partial product bits from the multiplicand and multiplier. PP reduction is used to compress the partial product bits to two. Finally the partial products bits are summed by using carry Propagate addition. The output is in the form of LSB and MSB. Finally the LSB part is compressed by using operations such as deletion, reduction, truncation, rounding and final addition.

PARTIAL PRODUCT GENERATION:

PP (partial product) generation produces partial product bits from the multiplicand and multiplier.

Example: Hexadecimal number format

a = 8d, b = 6c

Where ‘a’ is multiplicand and ‘b’ is multiplier.

Example:

							A=8D =>		1	0	0	0	1	1	0	1
							B=6C =>	X	0	1	1	0	1	1	0	0
									0	0	0	0	0	0	0	0
								0	0	0	0	0	0	0	0	x
							1	0	0	0	1	1	0	1	x	x
						1	0	0	0	1	1	0	1	x	x	x
					0	0	0	0	0	0	0	0	x	x	x	x
				1	0	0	0	1	1	0	1	x	x	x	x	x
			1	0	0	0	1	1	0	1	x	x	x	x	x	x
		0	0	0	0	0	0	0	0	x	x	x	x	x	x	x

PARTIAL PRODUCT REDUCTION:

Partial Product reduction is used to compress the partial product bits to two. Finally the partial products bits are summed by using carry Propagate addition.

Reduction Scheme1 and Reduction Scheme 2

Fig. 2 shows the reduction procedure of Scheme 1, reduction starting from the least significant column. Column height is h, including the carry bits from least significant columns, are also shown on the top row where the columns that need HAs are highlighted by square boxes.

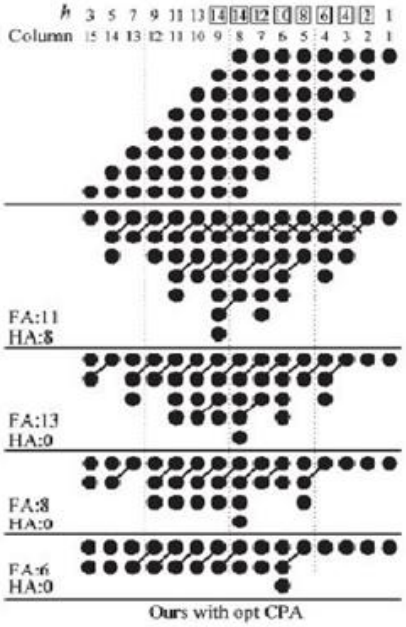


Fig.2shows the reduction procedure of scheme1 multiplier (38 FAs and 8 HAs)

These methods mainly discuss about the cost compensation. By comparing the methods of Dadda, Wallace with scheme1 and Scheme2, the reductionof bits are better, so the area can be saved higher than the former methods. From the literature survey it is clear that various researchers are working in these areas tooptimize the same. Compression ratio also takes up its major concern here, were it plays a crucial role when output precision is concerned. Fig.3shows the reduction procedures by scheme 2 to each column of partial product bits, reduction startingfrom the least significant column.

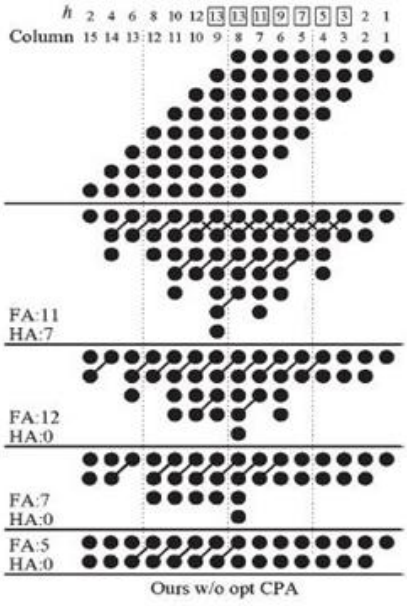


Fig.3 shows the reduction procedure of scheme2 multiplier (35 FAs and 7 HAs)

Scheme 1 having minimum CPA (carry propagate addition) bit width as twice reduction efficiency when compared to the Wallace method which produces the sameresult as that of RA method. Scheme 1 is only used to determine whether an HA is needed and how many FAs are required in the per-column reduction that does not exceed the maximum number of Carry Save Additions in reduction levels.

The scheme1, scheme2 and proposed multiplier architecture has been simulated and synthesized using XILINX ISE Design. From the synthesized results, the scheme 1 and scheme 2 has 1056 and 822 number of gates. The proposed multiplier has only 582 gates. Area utilization by the proposed method is less when compared to scheme 1 and scheme 2.

CARRY PROPAGATE ADDITION:

The partial products bits are summed by using carry Propagate addition.It is possible to create a logical circuit using multiple full adders to add *N*-bit numbers. Each full adder inputs a C_{in}, which is the C_{out} of the previous adder. This kind of adder is a ripple carry adder, since each carry bit "ripples" to the next full adder. Note that the first full adder may be replaced by a half adder.

The layout of a ripple carry adder is simple, which allows for fast design time; however, the ripple carry adder is relatively slow, since each full adder must wait for the carry bit to be calculated from the previous full adder. The gate delay can easily be calculated by inspection of the full adder circuit. Each full adder requires three levels of logic. In a 32-bit [ripple carry] adder, there are 32 full adders, so the critical path (worst case) delay is 3 (from input to carry in first adder) + 31 * 2 (for carry propagation in later adders) = 65 gate delays. A design with alternating carry polarities and optimized AND-OR-Invert gates can be about twice as fast.

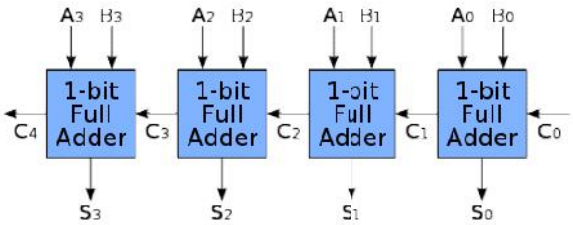


Fig.4. 4-Bit Ripple Carry Adder

III.PROPOSED TRUNCATED MULTIPLIER

The objective of a good multiplier is to provide a physically compact, good speed and low power consuming chip. To save significant power consumption of a VLSI design. In a truncated multiplier, several of the least significant columns of bits in the partial product matrix are not formed.

Fig.5 Shows 8x8 truncated multiplication. (a)deletion, reduction and truncation. (b) Deletion, deduction, truncation, and final addition.

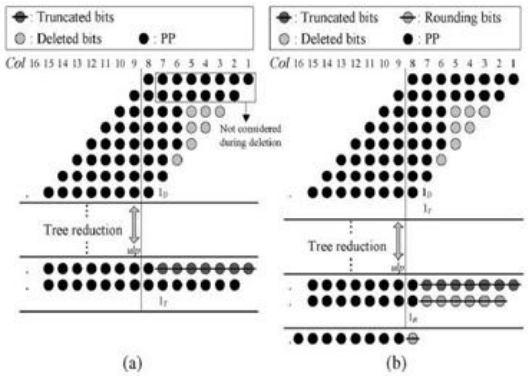


Fig.5 8x8 truncated multiplication. (a)deletion, reduction and truncation. (b)deletion, reduction, truncation, and final addition.

A. Deletion, Reduction, and Truncation of partial product Bits

In the first step deletion operation is performed, that removes all the avoidable partial product bits which are shown by the light gray dots (fig 5). In this deletion operation, delete as many partial product bits as possible. Deletion error $E_D$should be in the range $-1/2$ ulp$\leq E_D \leq 0$.Hereafter, the injection correction bias constant of ¼ ulp.

The deletion error after the bias adjustment $-1/4$ ulp$\leq E_D \leq 1/4$ ulp. In Fig..5, the deletion of partial product bits starts from column 3 by skipping the first two of partial product bits. After the deletion of partial product bits, perform column-by-column reduction of scheme 2.

After the reduction, perform the truncation, which will further removes the first row of (n-1) bits from column 1 to column (n-1). It will produces the truncation error which is in the range of $-1/2$ ulp $\leq E_T \leq 0$. Hence introduction of another bias constant of ¼ ulp in truncation part. So the adjusted truncation error is $-1/4$ ulp$\leq E_T \leq 1/4$ ulp.

B.Rounding and Final Addition

All the operations (deletion, reduction, and truncation) are done, finally the PP bits are added by using CPA (carry propagate addition) to generate final product of P bits. Before the final CPA, add a bias constant of ½ ulp for rounding. Rounding error is in the form of - 1/2 ulp $\leq E_R \leq$1/2 ulp. The faithfully truncated multiplier has the total error in the form of –ulp<E=(E_D+E_T+E_R) ≤ulp.

C. Proposed Algorithm

In proposed architecture we can multiply 8x8 bits, and the bits are reduced in step by step manner. Deletion is the first operation performed in Stage 1 to remove the PP bits, as long as the magnitude of the total deletion error is no more than 2^{-P-1}.Then number of stages to reduce the final bit width without increasing the error.

In normal truncated multiplier design, the architecture produces the output with some truncation error. But in the proposed design of truncated multiplier the truncation error is not more than 1 ulp, so the precision of the final result is improved. Fig. 5 shows proposed truncated multiplier.

This reduces the area and power consumption of the multiplier. It also reduces the delay of the multiplier in many cases, because the carry propagate adder producing the product can be shorter.

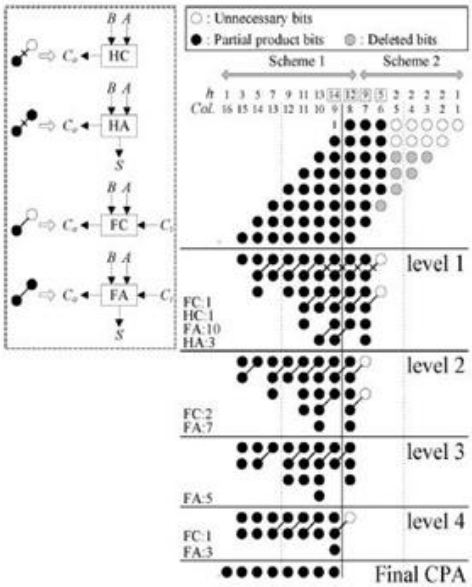


Fig.6 Proposed Truncated Multiplier.

IV. FUTURE SCOPE

Truncated multiplier can be effectively

implemented in FIR filter structure. Conventional FIR filer performs ordinary multiplication of co-efficient and input without considers the length.

Thus the structure can be made effective by replacing the existing multiplier with the proposed fixed width truncated multiplier for visible area reduction.

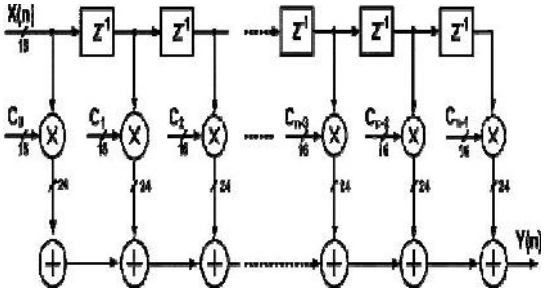


Fig7. Shows the architecture of FIR Filter.

THEORITICAL CALCULATIONS:

a= Multiplicand, b= Multiplier and P=Product

HexaDecimal format,

a= 89 = 1000 1001
b= a5 = 1010 0101

P = 0101 1001

In above result we had MSB part 8 bits only. So we need 16 bit result add LSB part equal to zero.

The result is 0101 1010 | 0000 0000.
MSB LSB

EXPERIMENTAL RESULTS:

By using the Synthesis tool is Modelsim. The proposed system is implemented by using FPGA-Spartan 3E.This methods are mainly applicable in DSP systems**.**

A. Power Analysis

TABLE 1 POWER ANALYSIS OF THE SCHEME 1, 2 & PROPOSED

Parameter	Scheme 1	Scheme 2	Proposed
Power(W)	0.185	0.176	0.088

The scheme1, scheme2 and proposed multiplier architecture has been simulated and synthesized using XILINX ISE Design Suite 8.1. From the synthesized results, it is found that the scheme 1 consumes 185mW, scheme 2 consumes 176mW. The proposed multiplier consumes low power of 88mW when compared to scheme 1 and scheme 2.

B. Area Analysis

TABLE 2 AREA ANALYSIS OF THE SCHEME 1, 2 & PROPOSED

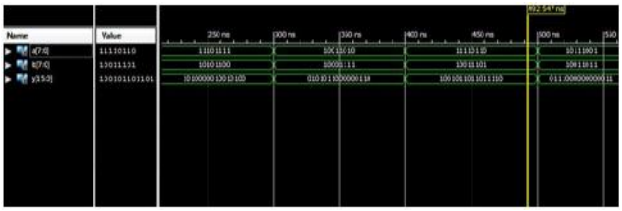
Parameter	Scheme 1	Scheme 2	Proposed
No. of Gate counts	1056	822	582

The table 1 & 2 shows that the proposed method reduces the power and area than the previous methods. When compared to previous methods the precision is improved.

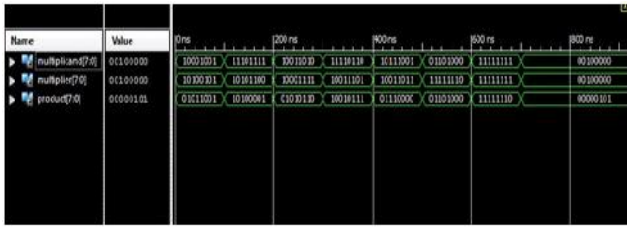
SIMULATION RESULTS:

Scheme1 Multiplier

Scheme2 Multiplier



Proposed Truncated Multiplier



CONCLUSIONS

There are many works proposed to reduce the truncation error by adding error compensation circuits so as to produce a précised output. In this approach jointly considers the tree reduction, truncation, and rounding of the PP bits during the design of fast parallel truncated multipliers, so that the final truncated product satisfies the precision requirement.

From the synthesized results, it is found that the scheme 1 consumes 185mW, scheme 2 consumes 176mW. The proposed multiplier consumes low power of 88mW when compared to scheme 1 and scheme 2. The scheme 1and scheme 2 has 1056 and 822 number of gates. The proposed multiplier has only 582 gates. Area utilization by the proposed method is less when compared to scheme 1 and scheme 2.

REFERENCES

[1] R. Devarani, PG Scholar, M.E VLSI Design "Design and implementation of Truncated Multipliers for precision improvement" 2013 International Conference on Computer Communication and Informatics (*ICCCI* -2013), Jan. 04 – 06, 2013, Coimbatore, INDIA.

[2] M. J. Schulte and E. E. Swartzlander, Jr., "Truncated multiplication with correction constant," in VLSI Signal Processing VI. Piscataway, NJ:IEEE Press, 1993, pp. 388–396 [3] Li-Rong Wang, Shyh-JyeJou and Chung-Len Lee, "A well-tructured Modified Booth Multiplier Design" 978-1-4244-1617-2/08/$25.00 ©2008 IEEE.

[3] J. M. Jou, S. R. Kuang, and R. D. Chen," IEEE Trans. Circuits Syst. II, s Analog Digit. Signal Process", vol. 46, no. 6, pp. 836–842, Jun. 1999.

[4] E. J. King and E. E. Swartzlander, Jr., "Data-dependent truncation scheme for parallel multipliers," in Proc. 31st Asilomar Conf. Signals, Syst. Comput., 1997, pp. 1178– 1182.

[5] L.-D. Van and C.-C. Yang, "Generalized low-error area-efficient fixed width multipliers," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 52, no. 8, pp. 1608–1619,Aug. 2005

[6] Shiann-RongKuang, Jiun-Ping Wang, and Cang-Yuan Guo, "Modified Booth multipliers with a Regular Partial Product Array," IEEE Transactions on circuits and systems-II, vol 56, No 5, May 2009.

G. RAKESH Pursuing, M.TECH in VLSI from JBREC College Affiliated from JNTUH. Moinabad, RangaReddyDist, Hyderabad,Telangana. He got 76.1% in his MTech 1st year. He has received the B.TechDegee in Sri KottamTulasi Reddy Memorial College of Engineering in 2011.

R.Durga Gopal, He obtained his M.Tech from CVR College of Engineering College, Hyderabad and pursuing Ph.D from JNTUH. Currently working as a Associate Professor in Joginpally B.R. Engineering College, His area of research includes Signal Processing, Communications and VLSI. In his career he guided so many B.Tech and M.Tech students to improve theirknowledge.

Dr. D.N Rao,B.Tech,M.E,Ph.D, Principal of JBREC, Hyderabad. His carrier spans nearly three decades in the field of teaching, administration, R&D, and other diversified in-depth experience in academics and administration. He has actively involved in organizing various conferences and workshops. He has published over 11 international journal papers out of his research work. He presented more than 15 research papers at various national and international conferences. He is Currently approved reviewer of IASTED International journals and conferences from the year 2006. He is also guiding the projects of PG/Ph.D students of various universities.