An Enhanced Low Power High Speed Dadda Multiplier using MTCMOS Technique

[1] G.Vandana, [2] A.Phanindra, [3] G.Mamamatha, [4] G.Satyavathi
[1]Assistant Professor, [2]UG Student, [3]UG Student, [4]UG Student
[1] Department of Electronics and Communication Engineering,
[1]WISTM, Visakhapatnam, Andhra Pradesh.

Received: January 27, 2019 Accepted: March 06, 2019

ABSTRACT: *Currently the tendency is to modify towards low area designs due to the increasing cost of scaled CMOS. In most of the industrial areas like image processing, DSP, microprocessor, it is required to do arithmetic operations very fast with greater precision and multiplier is widely used in these application regions. So we decide to increase the speed of multiplication unit. To improve the speed of multiplier we minimize the delays in multiplication and addition operations at every stage. The model of 4-bit multiplier using Algorithm named Dadda multiplier, which is used to improve the low power dissipation and minimum propagation delay. Full and half adder blocks have been designed using CMOS and MTCMOS process technology to reduce the power dissipation and propagation delay. The suggested Dadda multiplier utilizes 4:2 compressors in their partial product reduction stages*

Key Words: *Braun Multiplier, Dadda algorithm, MTCMOS, Wallace Tree Multiplier.*

I. Introduction

Multiplication is the basic process which is used in different electronic and in various digital communication applications [1]. Multipliers with low latency and minimum power dissipation are preferred to design an enhanced circuit so that maximum throughput can be achieved in minimum response time. Structure blocks used in multipliers are a full adder and a half adder. Different design implementations of a full adder and half adder circuits have been used to reduce power and delay in order to design an improved multiplier circuit which includes CMOS process technology. Besides this, different multiplication algorithms also have been used to achieve optimized power and delay product which includes Dadda, Wallace tree, Braun [2,3]. This design operates at high frequency and consumes less power as compared to preceding designs. In this paper, a multiplier has been implemented in which half adder and full adder has been exploited as its building block to reduce the power dissipation by using CMOS logic and MTCMOS technology process. Dadda Algorithm has been used to reduce the propagation delay and power dissipation of the multiplier.
This paper includes literature review in section 2, Dadda algorithm in section 3 and proposed enhanced MT-DADDA in section 4. Section 5 provides with simulation results and is compared with traditional multipliers such as Braun, Wallace tree and conventional DADDA algorithm in terms of power, delay, area and PDP and concludes in Section 6.

II. literature review

High speed multiplication is a key requirement of high performance digital systems. In recent advancements the column compression multipliers are popular for high speed calculations due to their higher speeds [1-3].

2.1. BRAUN MULTIPLIER

The simplest parallel multiplier is the Braun array [2,3]. All the partial products are computed in parallel, and then collected through a cascade of Carry Save Adders. The finishing time is restricted by the depth of the CSA, and by the carry propagation in the adder. Note that this multiplier is only suited for positive operands [3].

2.2 WALLACE TREE MULTIPLIER

The first column compression multiplier was presented by Wallace in 1964 [2, 3]. He abridged the partial product of N rows by grouping into sets of three row set and two row set using (3, 2) counters and (2, 2) counters respectively. In 1965, Dadda improved the approach of Wallace by starting with the exact placement of the (3, 2) counters and (2, 2) counters in the maximum critical path delay of the multiplier [3,4].

2.3. MTCMOS TECNOLOGY

As the feature size scales down, power consumption has become a primary design concern. Scaling of supply and threshold voltage has been achieved with transistor sizing that increases leakage power exponentially and has become the dominant factor in designing deep submicron circuits. Multi-Threshold CMOS (MTCMOS) technique is a promising and most widely used technique in controlling and minimizing high leakage technique that incorporates with two or more different threshold voltage transistors in a circuit during idle mode. Low-threshold voltage (LVT) cells have high speed but have high leakage currents, whereas high-threshold voltage (HVT) cells offer low speed but low leakage. MTCMOS incorporates these two types of cells where LVT cells are used for faster switching operations and HVT cells to gate the power supply (VDD) to down sub-threshold leakage significantly. There are several ways to realize MTCMOS in synchronous architectures. One technique is to exploit LVT cells for critical paths to achieve high performance and HVT cells for the non-critical paths to decrease leakage.

An approach for integrating multi-threshold concept into CMOS logic is fine-grained technique. This technique incorporates the MTCMOS method within every gate using LVT cells for the Pull-Up Network (PUN) and Pull-Down Network (PDN) and HVT cells to gate the leakage between the PUN and PDN networks. During sleep mode two additional LVT cells are incorporated in parallel with the PUN and PDN to retain equivalent potential across these two networks. When logic '1' input is given to the sleep signal, then the sleep cells between PUN and PDN will be off and cells that are in parallel to PUN and PDN will be on providing logic '0' output. When logic '0' is applied to the sleep signal, then the sleep cells between PUN and PDN will be on and cells that are in parallel to PUN and PDN will be off providing the valid output according to the circuit functionality. Thus this technique resolves the problems of sleep transistor sizing and logic block isolation resulting in large area overhead.

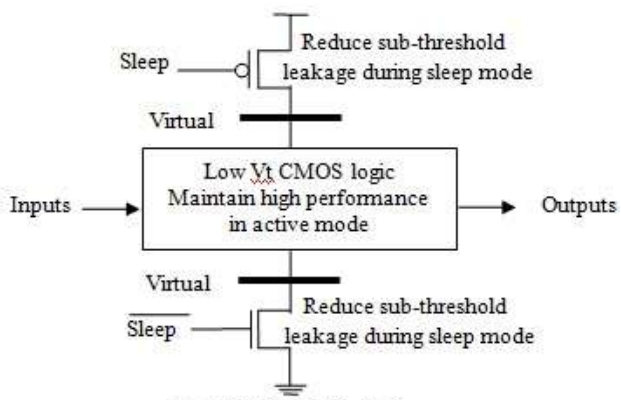

Fig.1 MTCMOS Technique

III. dadda algorithm

Since 2000's, a closer modification of Wallace and Dadda multipliers has been done and proved that the Dadda multiplier is slightly faster than the Wallace multiplier and the hardware required for Dadda multiplier is smaller than the Wallace multiplier. Then the Dadda multiplier attains faster switching, we implement the proposed techniques in the same and the improved performance is compared with the regular Dadda multiplier [4]. The total delay of the multiplier can be split up into three parts, they are Partial Product Generation (PPG), the Partial Product Summation Tree (PPST), and finally due to the Final Adder. Of these the dominant components of the multiplier delay are due to the PPST and the final adder. The relative delay due to the PPG is small. Therefore major improvement in the speed of the multiplier can be achieved by reducing the delay in the PPST and the final adder stage of the multiplier. In this work the delay introduced by the PPST is reduced by using two independent structures in the partial products[5].

Let, us assume the final two-rowed matrix height d1 = 2, based on d1 the successive matrix heights are obtained from dj+1 = 1.5 * dj, where j = 1,2,3,4,....., Rounding of fraction in this matrix height should be done down to least. i.e., 13.5 = 13(rounded). The matrix heights will be in this fashion 2, 3, 4, 6, 9, 13, 19, 28..... Finally the largest d_j should be obtained such that derived matrix height shouldn't exceed the Matrix overall height [6].

- In the first reduction stage, the column compression is to carried with the [3, 2] and [2, 2] counters such that the obtained reduced matrix height should not exceed dj.
- During the compression, the sum is to be passed to same column in the next reduction stage and the carry is to be passed to the next column.

- The above two steps are to be repeated until a final two-rowed reduced matrix is obtained.

The reduction of stages using Dadda algorithm is shown in Fig.2, Fig.3, Fig.4 and Fig.5.

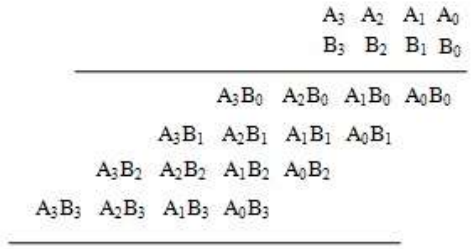

Fig.2 4X4 Multiplication

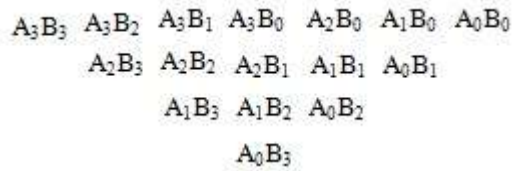

Fig.3 First Dadda Stage

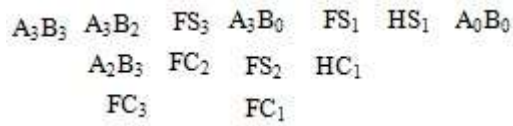

Fig.4 Second Dadda Stage

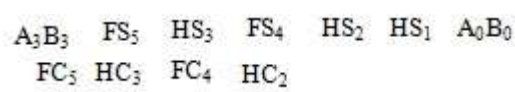

Fig.5 Third Dadda Stage

IV. proposed enhanced multi threshold dadda algorithm

Multi-threshold CMOS (MTCMOS) techniques is applied for Dadda Algorithm to enhance speed and to achieve low power. MTCMOS based dadda multiplier is shown in Fig.6 as it exploits LVT and HVT cells for faster switching and low leakage.

There are different ways to exploit MTCMOS in synchronous architectures. One method is to exploit LVT and HVT cells for critical paths and non-critical paths respectively to attain high performance and to decrease leakage.

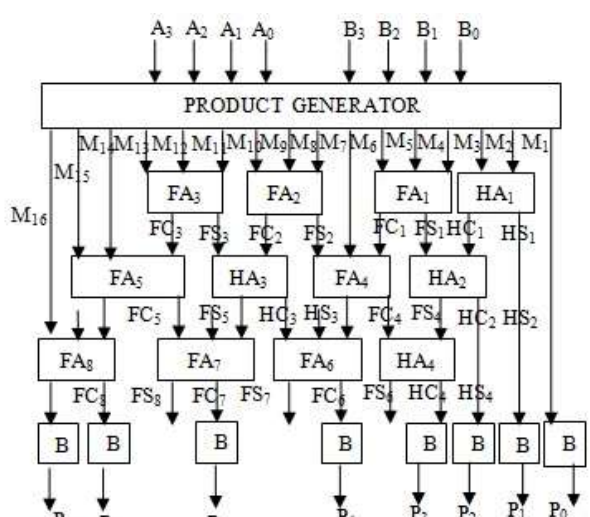

Fig.6 MT-Dadda Multiplier

IV. RESULTS AND DISCUSSION

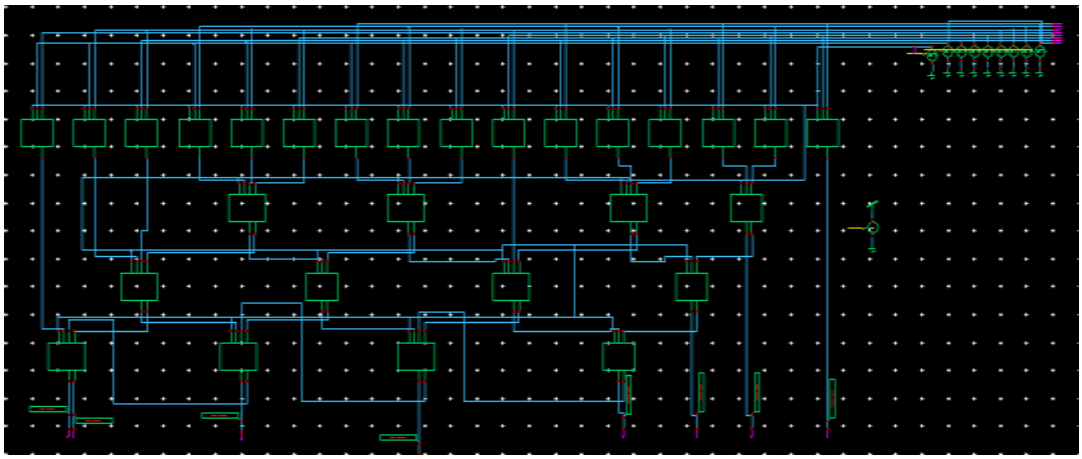

Fig.7 MT-Dadda Multiplier

Simulations are carried out using tanner EDA with 250nm technology. Evaluation results show that the proposed MTCMOS offers high speed and low power shown in Table 1. The conventional multipliers such as Braun, Wallace and Dadda Multiplier are compared with proposed MT-Dadda in terms of power, delay and PDP

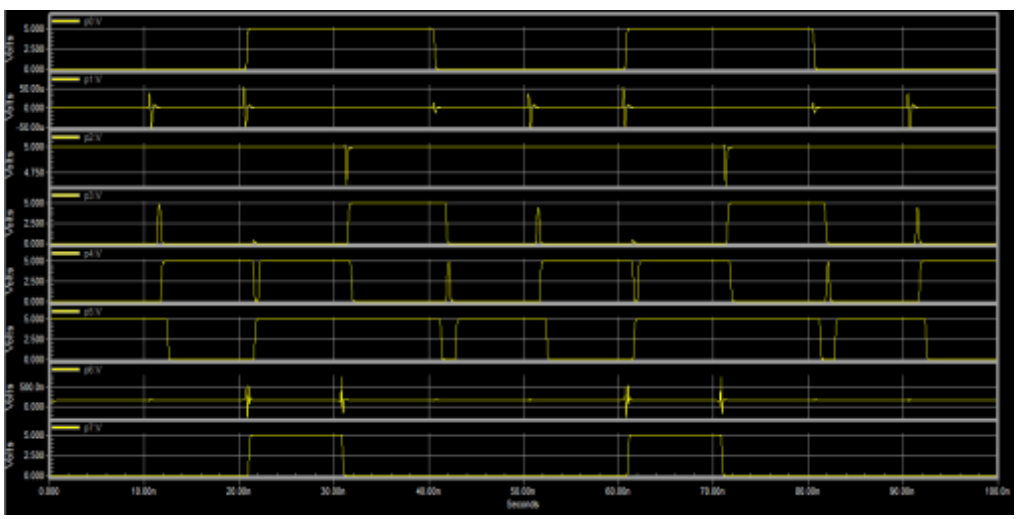

Fig.8 Output waveforms for MT-Dadda Multiplier

Fig.7 and Fig.8 shows the simulation waveforms for the proposed MT-Dadda algorithm.

Table 4.1: Comparative analysis of conventional multipliers with proposed enhanced MT-Dadda multiplier

Design	Power Dissipation (W)	Propagation Delay(ns)	PDP(Ws)
Braun Multiplier	$45.3*10^{-6}$	30.5850	$1.38*10^{-12}$
Wallace Multiplier	$41.4*10^{-6}$	30.5273	$1.26*10^{-12}$
Dadda Multiplier	$1.58*10^{-6}$	10.5417	$0.016*10^{-12}$
MT-Dadda Multiplier	$1.32*10^{-6}$	1.7330	$0.002*10^{-12}$

V. CONCLUSION & FUTURE SCOPE

The conventional and proposed MT-Dadda multipliers are designed by using Tanner EDA 250nm technology. The above table reports the PDP, power dissipation and propagation delay comes about for the traditional and the proposed multiplier. An altered 4x4 MT-Dadda tree multiplier has been composed and implemented. The outcomes scrutinizes that proposed design consumes less power and offers faster switching. The PDP for proposed structure is minimum. As a result the proposed enhanced MT-Dadda multiplier is most appropriate for high performance applications.

References

[1] Wallace C. (1964), A suggestion for a fast multiplier, IEEE Transactions on Electronic Computers, Vol. EC-13, pp. 14-17
[2] K.Roy and S.C. Prasad, Low Power CMOS VLSI Circuit Design (Wiley, 2000).
[3] Neil Weste and Kamran Eshranghian, Principles of CMOS VLSI Design (Addison Wiley, 2000).
[4] Luigi Dadda, Some Schemes for Parallel Multipliers, Alta Frequenza, Vol. 34, pp. 349-356, August 1965.
[5] W. J. Townsend, Earl E. Swartzlander and J.A. Abraham, A comparison of Dadda and Wallace multiplier delays, Advanced Signal Processing Algorithms, Architectures and Implementations XIII. Proceedings of the SPIE, vol. 5205, 2003, pages 552-560
[6] V. G. Oklobdzija and D.Villeger, Improving Multiplier Design by Using Improved Column Compression Tree and Optimized Final Adder in CMOS Technology, IEEE transactions on Very Large Scale Integration (VLSI) systems, Vol. 3, no. 2, June 1995.