International Journal of Combined Research & Development (IJCRD)
eISSN:2321-225X; pISSN:2321-2241 Volume: 2; Issue: 6; June -2014

DESIGN AND IMPLEMENTATION OF MULTIPLIER CIRCUIT FOR COMPLEX NUMBERS

NAVEEN S.M
Department of E and C,
R.V.COLLEGE of ENGINEERING
Bangalore -560059.
navee401@gmail.com

Ms. SUJATHA S.HIREMATH
Department of E and C
R.V.COLLEGE of ENGINEERING
Bangalore -560059.
sujathah@rvce.edu.in

Abstract : Complex number multiplication is an important arithmetic function in digital communication, radar, and optical system. Many major functions in the communication systems such as channel equalization; modulation and demodulation all deal with data streams and coefficients represented by complex numbers. Therefore efficient implementation of multiplier circuit for complex numbers is of significant importance. Consider two complex numbers, a=ar+jai and b=br+jbi and their product is P=ab=pr+jpi = (arbr-aibi) +j (arbi+aibr). The above equation shows that, the conventional implementation of this requires 4 multipliers, an adder, and a subtractor. This leads to a complex topology, since the interconnection becomes more complex, and also leads to high power dissipation. Here in this paper a new architecture has been introduced to reduce the complexity, and GDI technique has been used to reduce the power dissipation. This paper focuses two main design approaches. Initially the complex number multiplier is implemented using CMOS, and then it is compared with the GDI technique. The simulation results have shown that, the GDI consumes less power compared to CMOS. All designs were done using Cadence 90nm (GPDK 90) technology.

Keywords-Complex number multiplier, Gate diffusion input (GDI), complementary metal oxide semiconductor (CMOS), Complexity, Power, D Flip flop, Multiplexer, Multiplier, Adder/ Subtractor.

I. INTRODUCTION

The multiplier and adders are the essential elements of DSP (digital signal processors) and almost all communication circuits. In the field of digital design the binary multipliers are widely used. Since these multipliers are reliable and accurate to implement different types of operations. There are different types of multipliers based on the architecture. Based on the application particular multiplier will be selected.

The multiplier circuit consumes more power, so that the overall power dissipation of the circuit increases, which degrades the performance of the device. In case of multiplier the adder unit is the major source of power consumption, so improving the performance of the adder will increase efficiency of the multiplier and hence the final device.

In order to reduce the power consumption of a digital device, optimization of the digital circuit is required at every levels of the design. The optimization of any digital circuit is based on the technology used, architecture, and the logic style used to construct the circuits.

This paper presents design and implementation of multiplier circuit for complex numbers.

Consider two complex numbers say,

a=ar+jai and **b=br+jbi**

(1.1)

The complex product is

p=ab=pr+jpi= (ar.br-ai.bi) +j (ar.bi+ai.br).

(1.2)

As we can see from the equation 1.2, it shows that there are 4 multiplications, a subtraction and one addition. To implement this operation we need 4 multipliers, one adder and a subtractor. This consumes lot of circuit area. Since the area is one of the strong constraints, we have used a single multiplier in order to perform the 4 multiplications one after the other in sequence, and in order to separate the real and imaginary parts, an adder/subtractor is used. In order to store the intermediate and the final results the registers have been used. The entire operation completes over several clock cycles.

The Figure 1 shows the block diagram of the multiplier circuit for complex numbers. It is having multiplexers, multiplier; D flips flops, and an adder/Subtractor. And there are two types of signals (1).data signals and (2) control signals. The data signals are used to carry the data and the control signals, control the operation of various blocks of the complex number multiplier by enabling the appropriate block at the appropriate time.

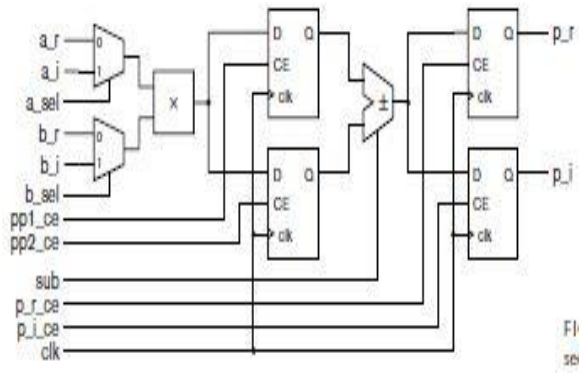

FIGURE 1: BLOCK DIAGRAM OF THE MULTIPLIER

The organization of the paper is as follows section describes the literature review, section 3 the design of individual blocks of the complex number multiplier. Section 4 describes the design of the proposed sequential complex number multiplier. Section 5 describes the simulation results; finally the conclusion is presented in section 6.

II. LITERATURE REVIEW

This paper, presents Modified Gate Diffusion Input technique used to implement radix 4 Booth Multiplier. Use of Booth algorithm as compare to other multiplication algorithms in multiplication process shows less computation and less complexity as it minimizes the total number of partial products to half of it. Booth multiplier at gate level can be design using any technique such as CMOS, PTL and TG but design with new MGDI technique gives better result in terms of area, switching delay and power dissipation.[1].

This paper presents the simplification of the addition operations in a low-power bypassing-based multiplier, a low-cost low-power bypassing-based multiplier is proposed. Row- bypassing multiplier, column-bypassing multiplier and bruan multipliers are implemented in conventional method and GDI technique. By optimizing the transistor size in each stage the power and delay are minimized. The results of post-layout simulation compared to similar reported ones illustrate significant improvement. Simulation results show great improvement in terms of Power-Delay-Product (PDP).[2]

This paper presents an evaluation of various adders that are representative of different CMOS logic design styles is carried out for nano scale CMOS technologies using a Predictive Technology Model from . The adders under consideration are the static CMOS mirror adder, the Complementary Pass-transistor Logic (CPL) adder, the Transmission Gate Adder (TGA), and the Hybrid- CMOS adder (HCMOS). The adders are evaluated in terms of delay, power dissipation, voltage scalability, and area. Because scaling of the voltage supply to less than 1 V results in gate overdrive factors of less than 0.5 V, it is found that adders that maintain the optimum signal paths for both high and low signals (the mirror and TGA adders) had the best performance metrics when scaled to the deep submicron regime.[3]

This paper, presents we shall introduce several new algorithms for integer multiplication that are based on specific multiple-radix representation of one of the multiplicands. We provide extensive theoretical analysis and experimental results for multipliers based on the new representations on 0:18 _m CMOS technology. They provide a clear picture about the advantages of the new method in 64-bit hardware implementations compared to array-based classical multiplier and radix-8-based multiplier. The proposed multipliers have better area and power consumption compared to reference multipliers.[4]

In this paper, a pulse-clocked double edge-triggered D-flip-flop (PDET) is proposed. PDET uses a split-output TSPC latch and when clocked by a short pulse train acts like a double edge-triggered flip-flop. The new double edge-triggered flip-flop uses only eight transistors with only one N-type transistor being clocked. Compared to other double edge-triggered flip-flops, PDET offers advantages in terms of speed, power, and area. Both total transistor count and the number of clocked transistors are significantly reduced to improve power consumption and speed in the flip-flop. The number of transistors is reduced by 56%-60% and the Area-Speed-Power product is reduced by 56%-63% compared to other double edge- triggered flip-flops In the literature survey discussed above, The GDI structure has advantages of low-power and high speed that is why we use GDI (GATE DIFFUSSION INPUT) adder in our paper.[5]

III. MULTIPLEXER DESIGN

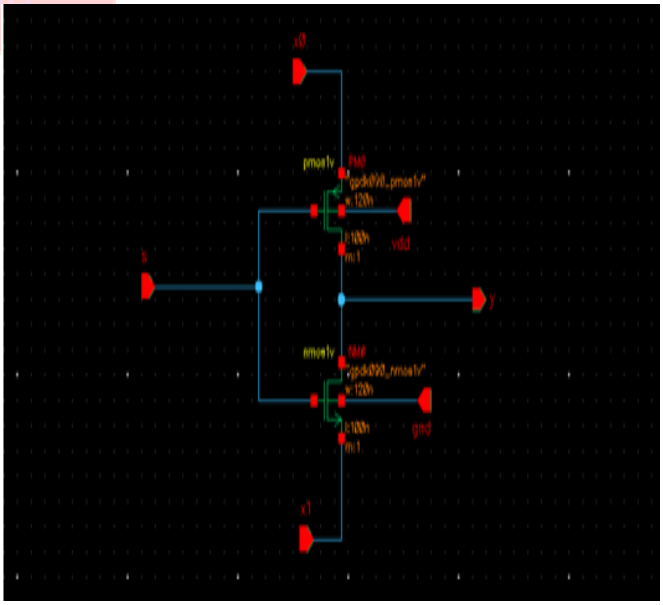

FIGURE 2: [2:1] MULTIPLEXER DESIGN.

Multiplexer means many into one. It is also known as data selector. Based on the select line, it will connect one of many inputs to output. For example a 2:1 multiplexer is shown in

FIGURE 2, has two data inputs, X0 and X1, a select input S, and an output Y. When the select input S is '0', output is connected to input X0 and when S='1', output is connected to input X1.

D FLIP FLOP WITH ENABLE DESIGN

In D flip flop the d input is transferred to output at every clock edges. But a D flip flop with enable works in somewhat different manner. The circuit diagram of D flip flop with enable is shown in FIGURE 3

As shown in the circuit diagram of FIGURE 3, it consists of a multiplexer at input of a positive edge triggered flip flop. Here when EN=1 the upper and gate is enabled, the D input is transferred to q output and when EN=0 the lower and gate is enabled, since the Q output is connected back to the D input of the positive edge triggered flip flop via lower AND gate and OR gate, the output continues with the previous state.

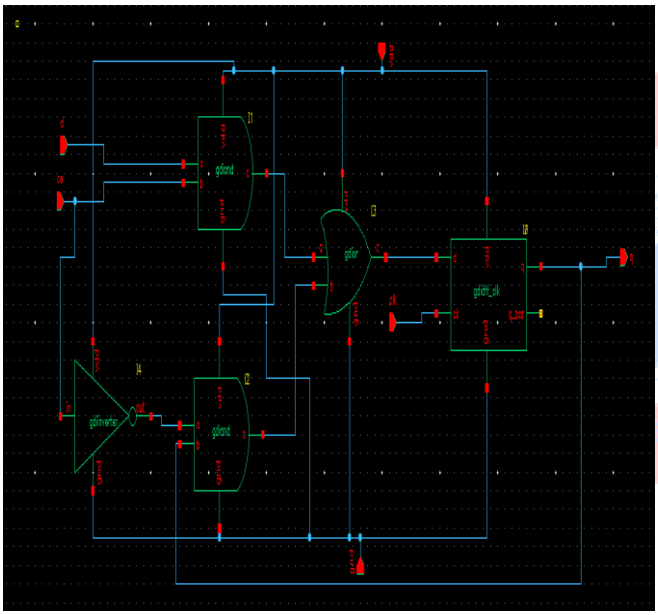

FIGURE 3: D FLIP-FLOP DESIGN

MULTIPLIER

Multiplier design can be divided into two blocks. These are

1. Partial Product Generation
2. Partial product Addition

The partial product generation is basically an AND operation of the I^{th} bit of multiplier with K^{th} bit of multiplicand. So we design an AND gate using GDI and CMOS logic design. For n-bit multiplication we require n^2 AND gates, hence for our design of 4-bit multiplier we require 16 and gates. The FIGURE 4 shows the design of a AND gate.

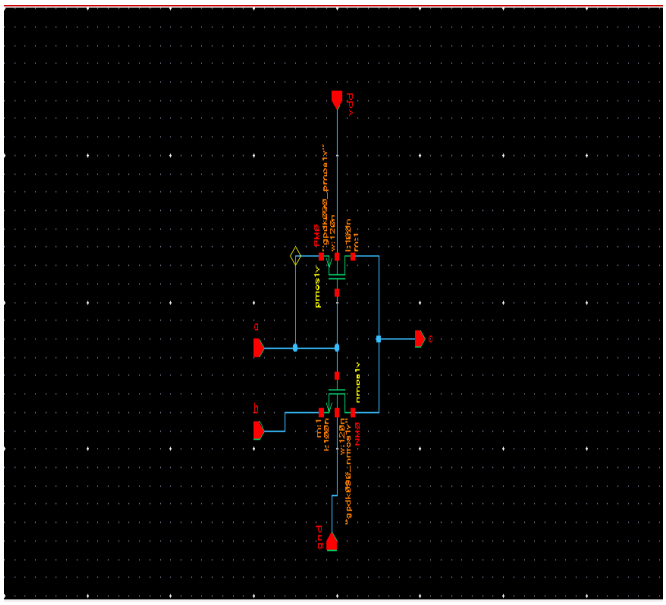

FIGURE 4: AND GATE DESIGN

PARTIAL PRODUCT ADDITION

The second step in the design of multiplier is to add the partial products generated in previous step. Various algorithms are present to add these partial terms, here we would use Wallace tree algorithm. The schematic of 3:2 Compressors is shown in FIGURE 5.

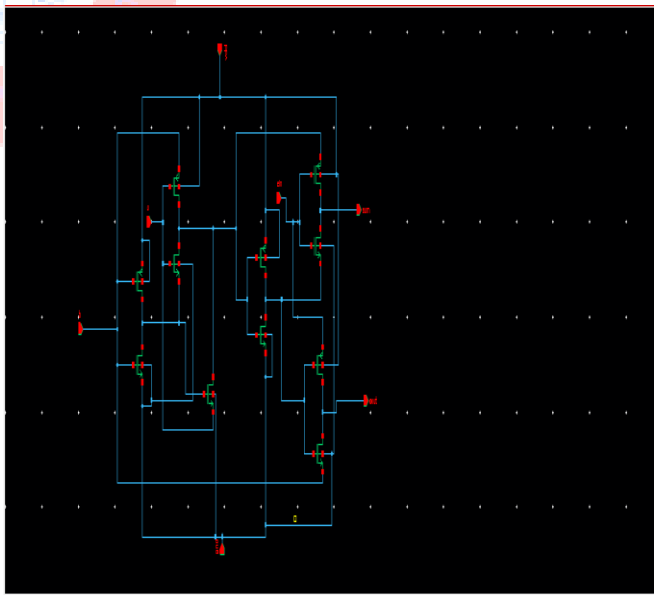

FIGURE 5: COMPRESSOR [3:2] DESIGN

The Wallace tree multiplier is designed using AND gates and adder, is shown in the FIGURE 6

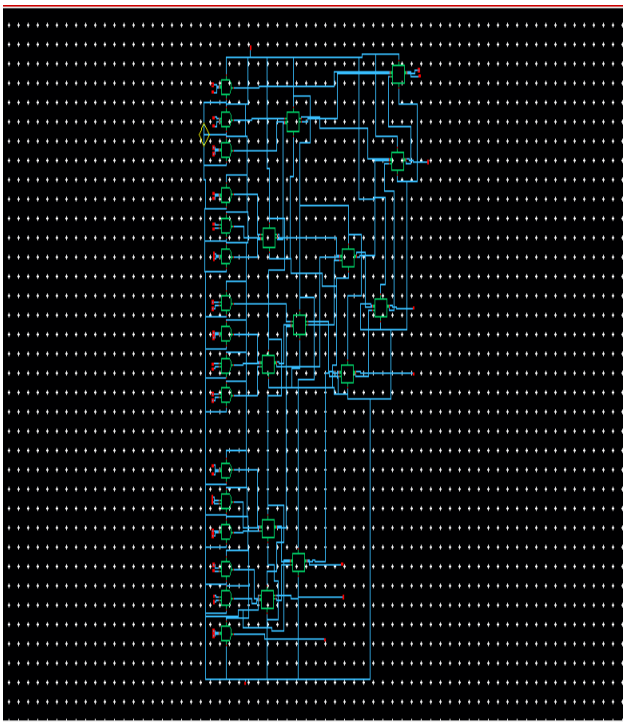

FIGURE 6: WALLACE TREE MULTIPLIER DESIGN

ADDER/SUBTRACTOR

The adder/Subtractor built using adders and XOR gates and add' sub control bit is shown in FIGURE 7. The adder circuit can be any adder to implement the adder/Subtractor. When add' sub=0, the XORs pass the Yi bits and the output is the sum (a+b). A control bit of add 'sub=1 changes the XORs into inverters, and the complemented values Yi' enter the full adders; add' sub=1 also acts as a carry in of c0=1. These operations combine to give the 2's complement algorithm for the difference (a-b). This technique is also applicable in a limited manner to other adder networks.

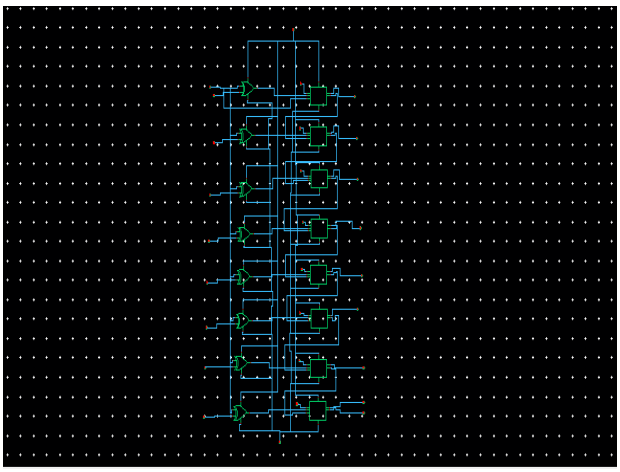

FIGURE 7: ADDER/SUBTRACTOR DESIGN

IV. DESIGN OF SEQUENTIAL COMPLEX NUMBER MULTIPLIER

The schematic of the multiplier circuit for complex numbers is shown in FIGURE 8. It consists of multiplexers at the input, multiplier; D flips flops, and an adder/Subtractor. And there are two types of signals (1).data signals and (2) control signals. The data signals are used to carry the data and the control signals control the operation of various blocks of the complex number multiplier by enabling the appropriate block at the appropriate time.

The operation of the complex number multiplier is as follows. Initially the operands are applied to the inputs of the multiplexers; the control signals A_SEL and B_SEL will select the particular data. This data is get multiplied by the multiplier. The result from the multiplier is stored in appropriate D flip flop by enabling the appropriate D flip flop by using the flip flop enable signals PP1_CE and PP2_CE. Then the adder/Subtractor unit either adds or subtract the data stored in D flip flops based on the control signal add_sub. When ADD_SUB=0, it will perform the addition and when ADD_SUB=1 then it will do the subtraction. This adder/subtractor is used to separate the real and imaginary parts. Finally the real part is stored in D flip flops which are reserved to store the real part. This is achieved by enabling the control signal PPR_CE at the right time. And the imaginary part is stored in another set of flip flops which are meant for imaginary results to store by enabling the control signal PPI_CE.

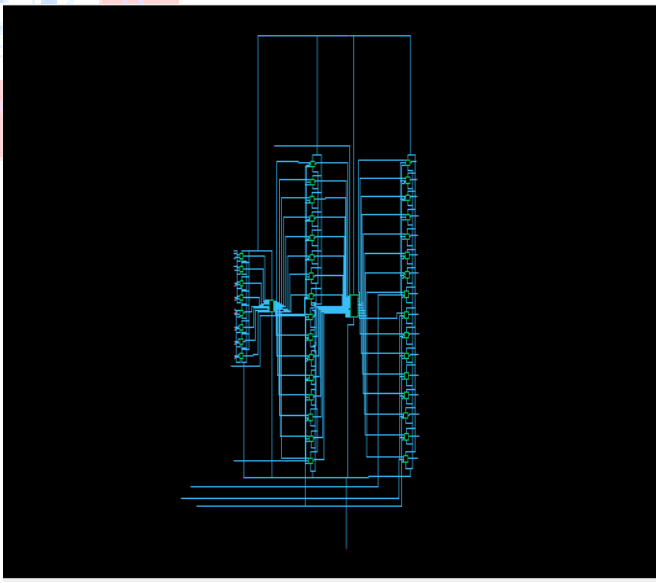

FIGURE 8:COMPLEX NUMBER MULTIPLIER DESIGN

V. SIMULATION RESULTS

The sequential complex number multiplier is verified for the inputs a=ar+jai=4+j3 and b=br+jbi=4+j3.The corresponding input and output waveforms of sequential complex number multiplier is shown in FIGURE 9.

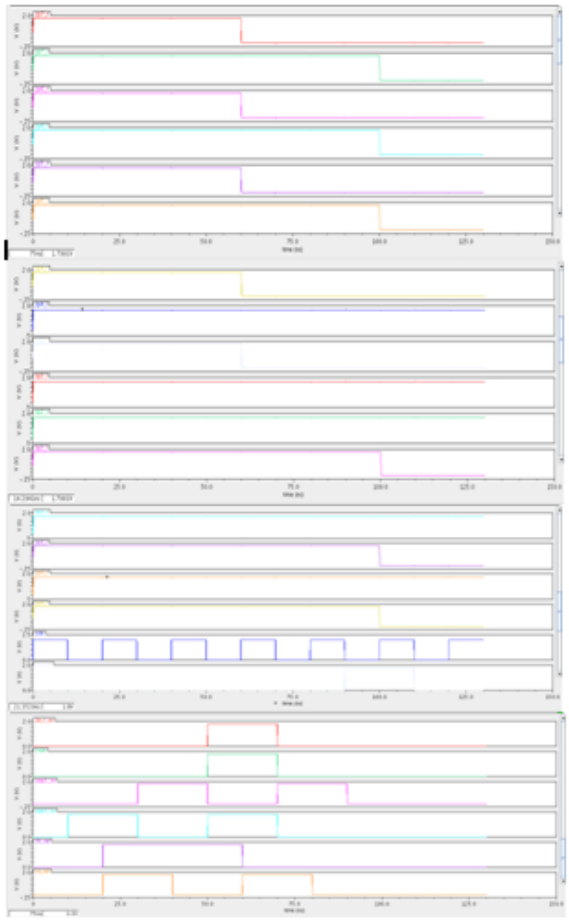

FIGURE 9: INPUT AND OUTPUT WAVE FORM OF SEQUENTIAL COMPLEX NUMBER MULTIPLIER

VI. CONCLUSION

This paper primarily was focused on the design of low power and high performance complex number multiplier. From TABLE 1 which shows that, Gate Diffusion Input (GDI), logic style used in this work provides us low power design as compared to CMOS logic styles. It also presents an area efficient approach to low power, as GDI requires less number of transistors as compared to CMOS for any design.

A multiplexer, 4-bit multiplier, Adder, Adder/subtractor, D flip-flop, and complex number multiplier were designed using GDI and CMOS logic styles in Cadence 90nm technology.

TABLE 1: POWER COMPARSSION OF COMPLEX NUMBER MULTIPLIER

Gate diffusion input complex number multiplier	CMOS complex number multiplier
$163X10^{-6}$ watts	$206X10^{-6}$ watts
$151.6X10^{-6}$ watts	$203.6X10^{-6}$ watts
$138.8X10^{-6}$ watts	$197.2X10^{-6}$ watts
$132.3X10^{-6}$ watts	$192.5X10^{-6}$ watts

REFERENCES

[1] Laxmi Kumre, Ajay Somkuwar, Ganga Agnihotri "Implementation of Radix 4 Booth Multiplier using MGDI technique" 978-1-4673-5149-2/13/$31.00 ©2013 IEEE

[2] Vamsi Krishna Rongali1, B Srinivas2," Design of Area Efficient High Speed Parallel Multiplier Using Low Power Technique on 0.18um technology", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 7, July 2013

[3] Rahul J. Gera and David H. K. Hoe, Member, IEEE," An Evaluation of CMOS Adders in Deep Submicron Processes", 44th IEEE University of North Florida, Jacksonville, FL March 11-13, 2012.

[4] Vassil S. Dimitrov, Kimmo U. Ja¨ rvinen, and Jithra Adikari, Student Member "Area-Efficient Multipliers Based on Multiple-Radix Representations" IEEE TRANSACTIONS ON COMPUTERS, VOL. 60, NO. 2, FEBRUARY 201[6] James E. Stine and Amey Phadke and Surpriya Tike "A Recursive-Divide Architecture for Multiplication and Division" 978-1-4244-9474-3/11/$26.00 ©2011 IEEE

[5] Bill Pontikakis, Mohamed Nekili,", A Novel Double Edge-Triggered Pulse-Clocked TSPC D Flip-Flop for High-Performance and Low-Power VLSI Design Applications", 0-7803-7448-7/02/$17.00 02002 IEEE.