International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2022): 7.942

Implementation of Low Power Multiplier for High Speed Arithmetic Applications

Kishor Kanaparthi

Department of Electronics and Communication Engineering, Vellore Institute of Technology, Andhra Pradesh., India

Abstract: ***In DSP and FPGA based applications power, speed and area all are important parameters and all are depend on multiplier which in turn depends on adders. So, by implementing adders we can reduce the delay. Pyramidaladders are used which uses half-adder and full-adder to increase the speed and to reduce the number of gates used in the multiplier, but delay is not decreased significantly. By using of modified half-adder and full-adder both gate count and delay is reduced normal to 16-bit multiplier. If we modify the Pyramidal adder with XNOR's and MUX instead of normal half-adder and full-adder, such pyramidal adder useless gates and delay is reduced compared normal 16-bit adder. The use of XNOR's and MUX in Pyramidal adder reduces delay, as the MUX function is only select the output among inputs. The use of such pyramidal adder in multiplier delay can be decreased greatly.***

Keywords: multiplexer, FPGA, XOR, Carry Generator

1. Introduction

In matrix multipliers, summation is carried by a matrix of adder, which consists successive rows of one – bit adders with transfer of saving, as the data moves transfer saving adds another partial product to the sum of partial products. High performance can be achieved with the irregularity, which is important when implementing such multipliers in integrated circuits [1]. Matrix multipliers like Braun multiplier use horizontal spreading of transfer whereas Baugh-Wooley's multipliers uses complementary codes for binary multiplication [7].

Tree-structured multipliers are faster than matrix multipliers however additional links are required which occupies more area in the crystal circuit compared to matrix multipliers [1], [5]. In multilayer multipliers summation is carried out by tree-like structured multiplier to compress partial products with same weight using one-bit half-adders and full-adders. Multilayer multipliers consume less power and less area, so very useful to DSP and high-performance systems.

In parallel multiplier add and shift algorithm is used, where partial products number to be added number is the main parameter and determines the performance of the multiplier. To reduce the number of partial products to be added modified Booth algorithm is used, however with increase in parallelism [8] the count of shifts between the partial products and intermediate sums will increase which may decrease speed and occupies more area for interconnecting [1]. In serial-parallel multiplier, speed is compromised to reduce power consumption and utility of allocated area. The selection of serial-parallel multiplier depends on type of application; Booth multiplier, Vedic multiplier and modified booth recoded multiplier are used for simulation. Binary half-adder is hardware formed complex circuit with five logic elements, if we use such half-adder in multi-combinational adders it gives more complex circuit For example, for 1024-bit DSP processor, and also speed is reduced due to serial connecting logic elements.For n*n bit multiplier, n*n AND gates and n (n-1) OR gates are required, in terms of adders n half-adders and n (n-1) full-adders are required[1], [3].

2. Literature Survey

Implementation of ALU and arithmetic devices depends on functionality of small bit processor (32- bit), accelerating methods are used for multi-bit processors(1024, 2048, …) with increase in complexity and irregularity in building patterns.For DSP and FPGA based multipliers accelerating methods must be used to reach the required speed.

2.1 Existing Method

Figure below shows the 4-bit matrix Braun multiplier, in which each column corresponds to the diagonal of the multiplier. Partial products formed with bits aj and bj using AND gates, to add partial products half adder and full adder are used. For Braun matrix n*n multiplier, n 2 products and n 2 -1 operations of 1-bit additions required where n is the bit capacity of the input data, speed of this multiplier is determined by longest route of signal processing i.e. –(3n-4).Tree-like structure Wallace's tree multiplier for n*n multiplier, n 2 products and n 2 -n operations of 1-bit additions required, so for full binary adding n 2 -n operations needed and for not full binary adding n operations required. Speed is limited by (2n-2) one-bit adders [1]. For small bit capacity multiplication Daddas [7] tree is widely used, n*n multiplier n 2 products and n 2 -n operations of 1-bit additions required where n is the bit capacity

2.2. Proposed Method

Improving the performance of digital adder is needed because execution of binary operation completely depends on adders, there are so many adders are implemented such that to meet the requirements of FPGA based VLSI environment and DSP Processor operations.

Ripple carry adder: It is simplest adder among all adders but slowest adder, it requires O (n) and delay of o(n), where n represent the operand size.

Carry look ahead adder: It has good area O (nlogn) and good delay of O(logn), but suffers from irregular layout design.

Carry select adder: It has area of O (n) and delay with O (n l+2/l+1), and it is the best adder in terms of area and delay
Carry save adder: Requires area O (n) and delay of O(logn).

Pyramidal adder: Hardware complexity of multipliers can be greatly reduced by using so many hardware structures and in those one of the structures is pyramidal adder. Pyramidal adder uses normal half-adder and full-adder. It contains single input 2n bit bus carrying inputs (a0b0, a1b1 a2b2 a3b3…..anbn), the pyramidal structures contains three single bit adders namely 2.1 block-to direct transfer of outputs, 2.2 block-to transfer inverse outputs, and 2.3block- to inverter the output bus of combiner adder (S0 S0, S1S2S3…..Sn), here 2.1 and 2.2 blocks acts as half adders as compared to half adder with five logic gates, so there is a reduction of gate count in the multipliers with 2.1 and 2.2 blocks. If binary Braun multiplier is implemented with pyramidal adder the gate count is reduced and speed of operation also increased [6].

For standard 4*4-bit Braun multiplier using normal half adders and full-adders consists there are 120 gates, if multiplier is implemented with pyramidal adder the gate count is 76, but we can get reduced delay compared to standard multiplier, then we will go for 16x16 bit pyramidal adder.

3. Implemented 16- bit Pyramidal Adder

16-bit Pyramidal adder implemented with modified half-adder and full-adder instead of using normal halfadder (2.1 block) and full-adder (2.2 blocks), this time for 16-bit pyramidal is implemented with both XNORs and MUX as shown in Figure 1, in this output selected from available inputs i.e. just selecting input, because of this implementation both gate count and delay are reduced compared to standard 16-bit adder which uses normal half adders and full adders.

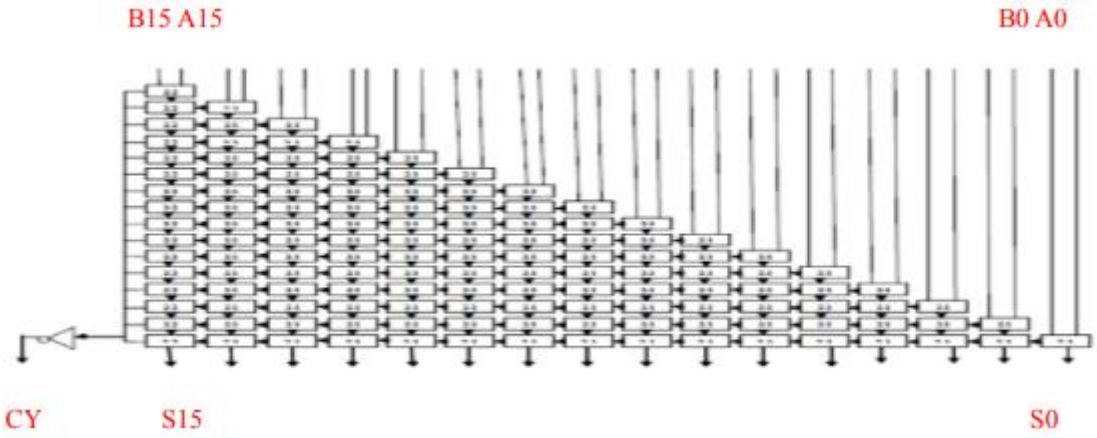

Figure 1: 16x16 – bit Pyramidal adder with modified half adder (2.1block) and full adder (2.2block**)**

This time for 16-bit pyramidal is implemented with both XNOR's and MUX , in this output selected from available inputs i.e. just selecting input , because of this implementation both gate count and delay are reduced compared to standard 16-bit adder which uses normal half adders and full adders.16-bit pyramidal adder consists two 16 bit inputs a and b respectively and they generate 16 sum output and 1 carry output, here 2.1 block and 2.2 blocks acts as half-adders hence each block generates one sum and one carry bit as shown in Figure 2

Figure 2: (a) Modified half adder (2.1 block) (b) Modified full adder (2.2 block)

4. Implementation of 16 – bit Multiplier using Modified Pyramidal Adder

16-bit Braun multiplier is implemented with modified half-adder and full-adder, for n*n multiplier if input a[15:0] and b[15:0] is applied we will get 2n output sum values i.e. S[31:0].In this Braun multiplier MUX selects the output among inputs, so delay is decreased significantly shown Figure 3.

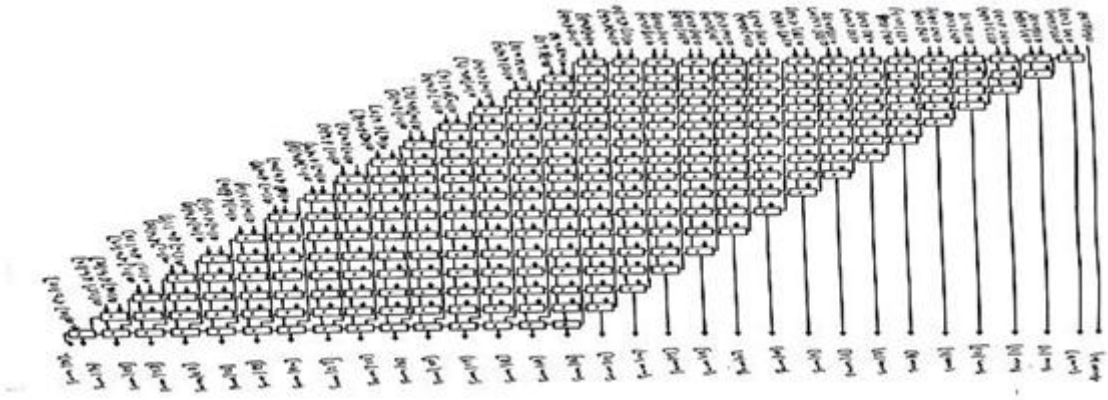

Figure 3: 16-bit Braun multiplier with modified half adder (2.1 block) and full adder (2.2 block)

Here this modified half adder acts as normal adder with less number of gates, so my using this modified halfadder the gate count can be reduced.

It contains single input 2n bit bus carrying inputs a0b0, a1b1 a2b2 a3b3…..a15b15, generates outputs as S0, S1, S2, S3, S4…..S31.

S0=a0&b0
S1= [(a2&b0) ^ (a0&b1)]
S2= [(a2&b0) ^ (a1&b1)]^ [c0^ (a0&b2)]
S3= [(a3&b0) ^ (a2&b1)]^ [c1^ (a1&b2)] ^ [c2^ (a0&b3)]
S4= [(a4&b0) ^ (a3&b1)] ^ [c3^ (a2&b2)] ^ [c4^ (a1&b3)]^ [c5^ (a0&b4)]
S5= [(a5&b0) ^ (a4&b1)]^ [c6^ (a3&b2)]^ [c7^ (a2&b3)]^ [c8^ (a1&b4)]^ [c9^ (a0&b5)]
S6= [(a6&b0) ^ (a5&b0)]^ [c10^ (a4&b2)]^ [c11^ (a3&b3)]^ [c12^ (a2&b4)]^ [c13^ (a1&b5)]^ [c14^ (a0&b6)]…
……

……

S28= [[(a15&b12) ^ (a14&b13)]^c229]^ [c230^ (a13&b14)]^ [c231^ ((a12&b15)]^ c232S29= [[(a15&b13) ^ (a14&b14)]^c233]^ [c234^ (a13&b15)]^ [c234^ ((a13&b15)]^ c235S30= [[[(a15&b14) ^ (a14&b15)] ^c236] ^ c237
S31= [(a15&b15)]^c238

5. Simulation results of 16 – bit Pyramidal Adder and 16 – bit Multiplier

Simulation and synthesis results of both 16 bit pyramidal adder and 16-bit multiplier are obtained using xilinx ISE 14.7 design software.

Figure 4: RTL analysisof16 – bit pyramidal adder

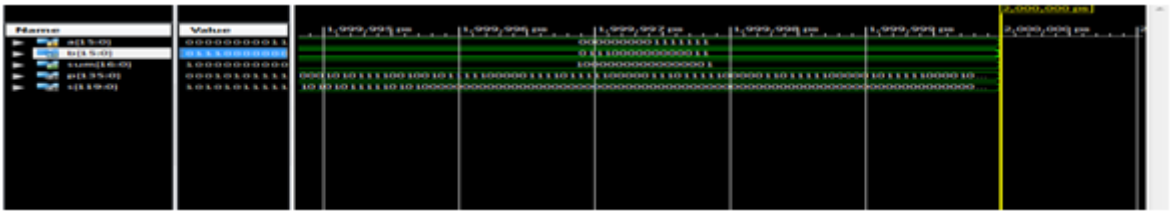

Figure 5: Graphical representation of16 – bit Pyramidal adder

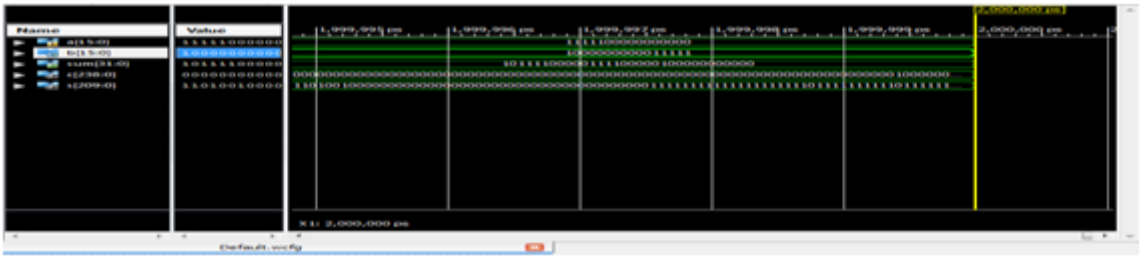

Figure 6: Graphical representation of 16x16 bit Braun multiplier

6. Conclusion

Power, area, speed and delay depend on multipliers which in turn depend on adders, so by implementing adders we can reduce the required parameter values. There are so many adder structures are there for binary multipliers, pyramidal adder is the one structure to reduce hard ware complexity and delay.

References

[1] V.Grega, I.Kolosov, O.Danyluk The development of a fast iterative algorithm structure of cosine transform in proceedings of XIIIth international conference TCSET2016—Lviv-Slavsko, Ukraine 2016-pp.506-509.
[2] S. Pezaris A40-ns 17b by 17b array multiplier, IEEE transactions on computers, c-20.Apr.1971.pp.442- 447.
[3] B. krulik Ovskyi, A.Davletova V. Gryga.Y.Nykolaichuk synthesis of components of high performance special processors of execution of Arithmetic and logical operations data processing in theoretical and numerical basis '' The Experience of designing and applications of CAD systems in microelectronics. proceedings of XIV th inernational conference MEMSTECH 2017-Lviv-Svalaya, ukaraine, 2017-p.170 -173.
[4] C.Baugh, Bwooley A 2's complement parallel array multiplication algorithm, IEEE transactions on computers, c-22.dec-1973, pp. 1045 – 1047.
[5] C.Wallace a Suggestion for parallel multipliers // IEEE Trans.Electron computers.-1964-v.13 –p. 14- 17. [6]. L.Dadda Some Schemes for parallel multipliers //Altra Freguenza-v.34 –p.349 356.
[6] Volodymyr Gyrga, Bogdan Dzunda, Ivan Dadaik , Yarslov Nykolaichuk Research and implementation of hardware Algorithm for multiplying binary numbers, 2018 IEEE , PP. 1277 -1281.
[7] V. Gryga Y. Nykolaichuk, N. Vozna, B.krulikovskyi synthesis of of a microelectronic structure of a specialized procesor for sorting an array of binary multipliers //Perspective technologies and nethods in MEMS design proceedings of inernational conference MEMSTECH 2017-Lviv-Svalaya, ukaraine, 2017-p.170 -173.

Author Profile

Kishor Kanaparthi received B-TECH degree from, Sri Sarathi Institute of Engineering& Technology, Nuzvid. Obtained M-TECH degree from JNTUA College of Engineering, Anantapur. Currently working as Technician at Vellore Institute of Technology, Andhra Pradesh