International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

HYBRID MAC IMPLEMENTATION USING BOOTH AND WALLACE APPROXIMATE ADDERS AND MULTIPLIERS

Mr.MVD PAVAN KUMAR[1], D.N.V.S. VIJAYA LAKSHMI[2]

[1]Assistant Professor, International School of Technology and Sciences for Women, Rajanagaram, Andhra Pradesh-533294.

[2]Associate Professor, International School of Technology and Sciences for Women, Rajanagaram, Andhra Pradesh-533294.

ABSTRACT:

Low power consumption and smaller area are some of the most important criteria for the fabrication of high performance systems. Optimizing the speed and area of the multiplier is a major design issue. However, area and speed are usually conflicting constraints so that improving speed results mostly in larger area. In this research main aim was to determine the best solution to this problem by comparing radix-4 multiplier using hybrid adder with normal radix-4 booth multiplier. Continuous advances of microelectronic technologies make better use of energy, encode data more effectively, reduce power consumption, etc. Particularly, many of these technologies address low-power consumption to meet the requirements of various portable applications. In these application systems, a multiplier is a fundamental arithmetic unit and widely used in circuits.

Keywords: MAC, CSA, CLA, RISC, CISC, FFT, CSLA, RCA.

I INTRODUCTION

The demand for high speed and efficient processing has been mounting as a result of growing computer and signal processing applications. The core of every processing system is its data path. Available statistics [3] has given clear indications that more than 70% of the instructions usually perform arithmetical and logical operations mainly consist of addition and multiplication in the data path of RISC and CISC machines. Multiplication based computation, which involve operations like Multiply and Accumulate and inner product most intensive arithmetic functions, currently implemented in many signal processing applications such as convolution, fast Fourier transform, filtering and in microprocessors in its arithmetic and logic unit. Since multiplication dominates the execution time of most signal/ instruction processing algorithms, so there is a need of speed efficient multiplier. Also, low power consumption and area efficiency are among the most important criteria for the fabrication of any processing and high performance systems. Optimizing the speed and area of the multiplier is a major design issue. However, area and speed are usually contradictory, so that improving speed results mostly in larger areas also. In our study, we propose the best solution to this problem by introducing a new efficient VLSI architecture of parallel Multiplier-and Accumulator (MAC) using hybrid approach for high-speed arithmetic operations. Since the accumulator that has the largest delay in MAC, the Carry Save Adder (CSA) and compressor techniques are used as one of the processing element to improve the overall performance. As per in the previous work, the general Booth algorithm and CLA's are used in MAC operation for getting the efficient output

results through pipeline scheme. But the problem rises by the general booth algorithm and CLA's in previous work by that pipeline scheme performance will be decreased. To rectify these problems this proposed method has been introduced the modified booth algorithm and CSA. This study presents an efficient implementation of high speed multiplier, Radix-8 modified Booth multiplier algorithm. The parallel multipliers like radix 2 and radix 4 modified booth multiplier does the computations using lesser adders and lesser iterative steps. However, the fact remains that the area and speed are two conflicting performance constraints. Hence, innovating increased speed always results in larger area. In this, we arrive at a better trade-off between the two, by realizing a marginally increased speed performance The proposed MAC will show the better properties to the existing standard design in many ways and performance twice as much as the previous research in the similar clock frequency.

Recent advances in communication and multimedia systems have resulted in a demand for efficient and fast digital signal processing systems. DSP systems and algorithms are used for processing data streams almost everywhere and therefore they require high precision and timing accuracy. High-performance digital signal processors consume very low power while operating at high speeds. Filtering, convolution, polynomial evaluation, and dot-matrix operations are some of the processes involved in processing signals digitally. These operations usually involve multiplication and addition and are performed by the Multiply and Accumulate (MAC) unit. The MAC Unit is an integral part of all Digital Signal Processors. Fast Fourier Transform (FFT) and DTFT require a large number of multiplication and addition operations. The performance of the processor depends largely on the MAC unit. The area occupancy, power consumption, and delay incurred in the MAC unit can influence the overall performance of the system. A MAC unit consists of an adder, a multiplier, and an accumulator. In general, the delay incurred in DSP systems is mainly due to long multiplication processes in the multiplier. A wide variety of multipliers are known to be in existence. Each multiplier has a unique structure and follows a unique algorithm. Vedic, Booth, array and Wallace tree multipliers are some of the multipliers that are being widely used. Analyzing and comparing these multipliers based on power consumption, the delay incurred and area occupancy can help identify the optimum multiplier to be used in the MAC unit.

II SURVEY OF RESEARCH

In this paper [1], the asynchronous floating-point arithmetic units consisting of adders/subtractors and multipliers are designed and compared based on the Balsa synthesis system. For the critical mantissa multiplication in the multiplier, the modified Booth algorithm (radix 2, 4, and 8) is adopted. A pipelined design of the multiplier is also presented to increase performance. Proposed MAC showed the superior properties to the standard design in many ways and performance twice as much as the previous research in the similar clock frequency.

In this paper [2] a new MAC architecture to execute the multiplication accumulation operation, for digital signal processing and multimedia information processing efficiently was proposed. By removing the independent accumulation process that has the largest delay and merging it to the compression process of the partial products, the overall MAC performance has been improved almost twice as that of the previous

work. Proposed both radix- 4 and Radix-8 Booth encodings new Radix- 5 kogge stone adder is developed to reduce the delay. Also some measures should be taken which minimize the area consumption.

In this paper[3]study of various parallel MAC architecture and then implementation of design of parallel MAC based on some Booth encodings such as Radix -4 Booth encoder and some final adders. Such as CLA, Kogge stone adder is carried out. Presented an efficient implementation of high speed multiplier using the shift and adds method, Radix-8 modified Booth multiplier algorithm. The architecture includes a final adder with the size of 2 to perform a multiplication. It means that the operational bottle neck is induced in the final adder no matter how much delay. Proposed for high speed and low power. For improving the speed and to reduce the dynamic power there is a need to reduce the glitches 1-0 transition and spikes 0- 1 transition. Adder designed using spurious power suppression technique (SPST) which avoids the unwanted glitches and spikes. A new MAC architecture to execute the multiplication accumulation operation, which is the key operation, for digital signal processing and multimedia information processing efficiently, was proposed. By removing the independent accumulation process that has the largest delay and merging it to the compression process of the partial products, the overall MAC performance has been improved almost twice as much as in the previous work.

In this paper[4], a new architecture for a high speed MAC, in which computations of multiplication and accumulation are combined and hybrid type CSA structure is used to reduce the critical path and improve output rate is achieved Present an efficient implementation of high speed multiplier using the shift and add modified Booth algorithm. The adder used is look ahead carry adder. The compression tree along with the carry look ahead adder has reduced the hardware overhead and power consumption. Proposed architectures of the high-speed low power and less area of modified Booth Wallace MAC. CSLA has comparatively low value of critical path length hence less combinational path delay but it has higher no. of leaf cell count and combinational path area.

It also has high dynamic power than CLA and CSKA. So CLA and CSKA architectures can be used for low power applications as it has low value of dynamic as well cell leakage power. A new multiplier Accumulator architecture based on high accuracy modified Booth algorithm. In this paper, a new MAC architecture is developed for high speed performance. The performance improvement is achieved by merging CSA and accumulator. MAC architecture is synthesized with 180 nm standard CMOS library using cadence SOC encounter. Multipliers with high speed are essential of digital applications for example signal processing. A new architecture of multiplier-and-accumulator (MAC) was proposed for high-speed arithmetic. By combining multiplication with accumulation the performance was improved.

In Modified booth algorithm technique the modified booth encoder will reduce the number of partial products. Even in general purpose processors high speed multipliers are most required to provide a physically compact, good speed and low power consuming chip. To save significant power consumption of a VLSI design, it is a good direction to reduce its dynamic power. This paper proposes the spurious power suppression technique (SPST) in VLSI will reduce

the power consumption of the system significantly.

In this paper [7], a new architecture of multiplier-and-accumulator (MAC) for high-speed arithmetic proposed. By combining multiplication with accumulation and devising a hybrid type of carry save adder (CSA), the performance was improved. The proposed CSA tree uses 1's-complement-based radix-2 modified Booth's algorithm (MBA) and has the modified array for the sign extension in order to increase the bit density of the operands. The CSA propagates the carries to the least significant bits of the partial products and generates the least significant bits in advance to decrease the number of the input bits of the final adder.

EXISTING METHODS-MULTIPLERS:

In digital processing systems, a fast multiplier is one of the most essential parts. In the early days, multiplication operation required two basic steps i.e. a shifting and addition in series. A multiplier determines the no of addition of multiplicand. All the addition output generates partial products. As the no of bits increases for both multiplicand and multiplier repeated addition sequences also increase which slows down the generation of end product generation. In order to remove this problem, many algorithms have been introduced already. A multiplier has got major two parts, the first part is partial products generator and the second one is accumulation and addition. For making a power & area efficient, high speed multiplier GDI technique has been introduced in these building blocks of a multiplier.

TYPES OF MULTIPLIER

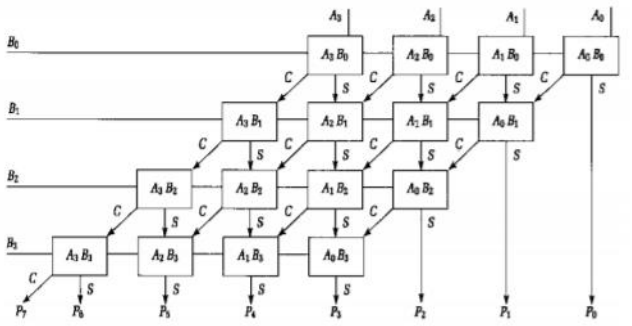
A. Serial Multiplier

Serial multiplications of two n bit numbers are the addition of n binary numbers, rows 1, 2, 3 upto n. One adder is used to add all the P x Q partial products. This type of multipliers show excellent power and area utilization but suffers in speed. The circuit is shown in the fig. below for 8 bit multiplication. It requires synchronization in taking of inputs as Multiplicand and Multiplier depending upon the length of the multiplicand and the multiplier. Therefore it requires different clocks. In this type of multiplier the delay is directly proportional to the length of multiplier & multiplicand. Such multiplication algorithm is not suitable for higher value of multiplier or multiplicand as here in this case all partial products are generated individually.

B. Array Multipliers

Array multiplier is very popular as having a regular structure. This type of multiplier is consists of adder and shifter blocks. Here, bitwise multiplication takes place. As per the position of the bit in the multiplier which is required to be multiplied with the multiplicand, each of the products is placed (shifted left). The actual result is 2N bits wide for an NxN multiplier, generated by getting the added value of all the corresponding products [16]. As this type of multiplier is an array of one bit multiplier block it suffers both in area as well as speed. To improve these simple adder blocks can be replaced by Carry Save Adders followed by ripple carry adder.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362



MAC Unit Due to advent use of computers and microprocessors analog signal processing is being performed in digital domain with the help of various DSP algorithms like DFT, FFT as the digital domain is less prone to error and more tends to accuracy. These algorithms have two basic operations i.e. multiplication and accumulation. By reducing the partial product generation during multiplication improves the speed of the MAC. A conventional MAC unit consists of a multiplier and an accumulator block along with adder network, the output from the multiplier as added to the value saved in accumulator with the help of adder network. AB+C are the function which is implemented in a conventional multiplier and accumulator unit.

III RECOMMENDED SYSTEM

Hybridization has been accomplished by undulating the third convey of a 4-bit CLA, through a full adder as in Figure. This strategy brought about decrease of land yet at the expense of expanded basic way delay. Further, the idea has been stretched out to have two and three full adders undulated in a similar way.

The Conventional CSA using CLA

CSA appeared in Figure is utilizes 4 bit CLA squares. Since the extent of the CLA squares can't be expanded, variable length can't be accomplished. Subsequently this is anything but a square root CSA outline.

Proposed hybrid architecture I

The principle issue in CSA utilizing CLA is, the quantity of bits registered per organize is restricted to four bits. In these new cross breed models the quantity of bits figured per MUX stage can be expanded. As appeared in Figure 3 CLA is utilized in introductory stages, HRCLA is utilized in 4 phases. In Later stages this technique lessens the including segments, containing just swell adders. By utilizing swell adders, the bits figured per MUX stage can be expanded straightly, prompting expanded speed.

Proposed hybrid architecture II

The second design depends on the hybridization of CSA by utilizing both RCA and CLA as in Figure 4. In this design the 4-bit CLA has been held in each stage. RCA is fell aggregately by one piece to the CLA in each stage (crossover structure). The quantity of phases of mux can be lessened further, when contrasted with the engineering I, thus the basic way delay has been additionally decreased.

4-bit adders with carry look ahead

A carry-look forward viper (CLA) or quick snake is a sort of snake utilized in advanced rationale. A convey look forward viper enhances speed by lessening the measure of time required to decide convey bits. It very well may be appeared differently in relation to the more straightforward, yet normally slower, swell convey viper (RCA), for which the convey bit is figured close by the entirety bit, and each piece must hold up until the point that the past convey bit have been ascertained to start computing its own particular outcome and convey bits. The convey look forward viper figures at least one convey bits

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

before the total, which lessens the hold up time to ascertain the after-effect of the bigger esteem bits of the snake. The Kogge– Stone viper (KSA) and Brent– Kung snake (BKA) are precedents of this kind of adder.

Booth Algorithm

Booth augmentation calculation or Booth calculation was named after the innovator Andrew Donald Booth. It very well may be characterized as a calculation or technique for increasing paired numbers in two's supplement documentation. It is a basic strategy to duplicate twofold numbers in which increase is performed with rehashed expansion tasks by following the stall calculation. Again this stall calculation for duplication activity is additionally changed and consequently, named as altered corner calculation.

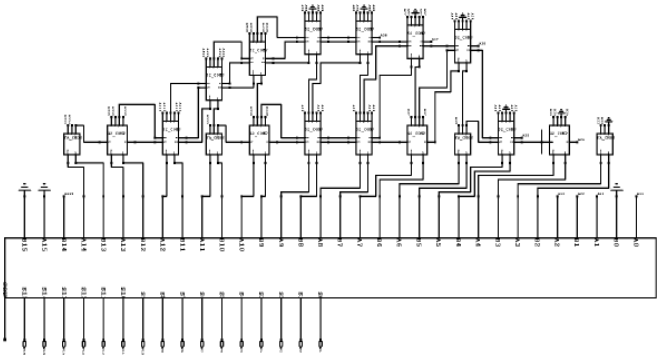


Fig.1. Proposed model.

In this area, fundamental MAC activity is presented. A multiplier can be separated into three operational advances. The first is radix-2 Booth encoding in which an incomplete item is created from the multiplicand (X) and the multiplier (Y). The second is snake exhibit or incomplete item pressure to include every single fractional item and convert them into the type of total and convey. The latter is the last option in which the last increase result is delivered by including the aggregate and to convey.

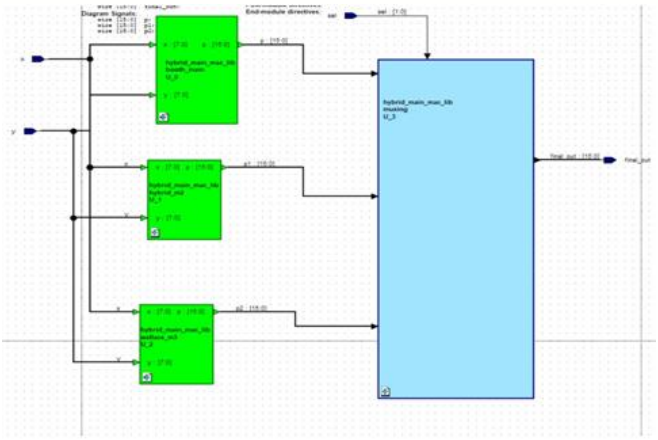


Fig.2. Proposed Design Block Diagram.

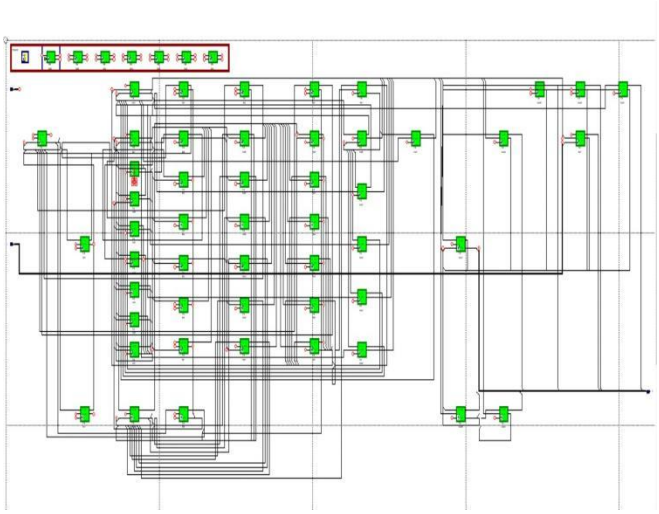


Fig.3. wallace multiplier flow diagram.

WALLACE ADDER structure is proposed to lessen the basic way and enhance the yield rate. It utilizes MBA calculation in light of 1's supplement number framework. An altered exhibit structure for the sign bits is utilized to build the thickness of the operands. A convey look-ahead viper (CLA) is embedded in the WALLACE ADDER tree to decrease the quantity of bits in the last snake. Likewise, with a specific end goal to build the yield rate by streamlining the pipeline effectiveness, middle of the road computation results are collected as entirety and convey rather than the last snake yields.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

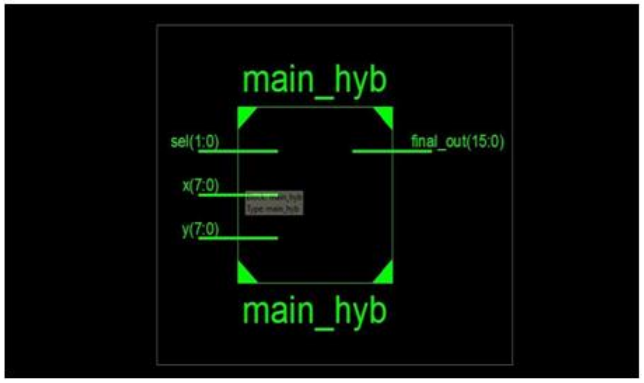


Fig.4. RTL Schematic diagram.

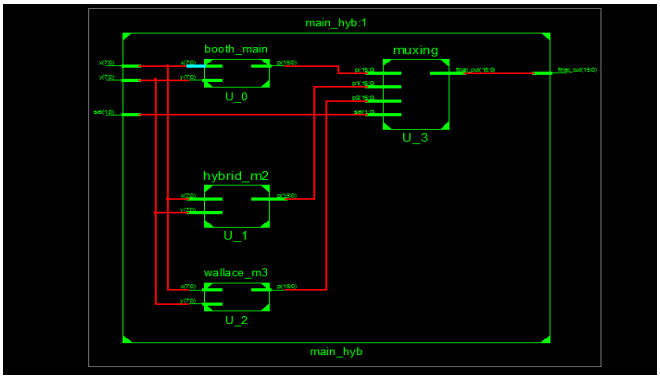


Fig.5. RTL Schematic diagram.

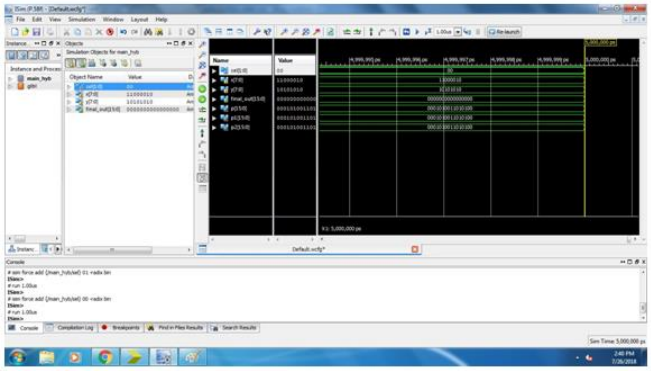
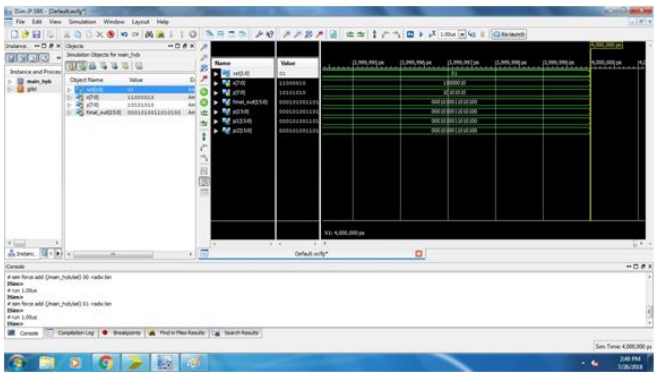
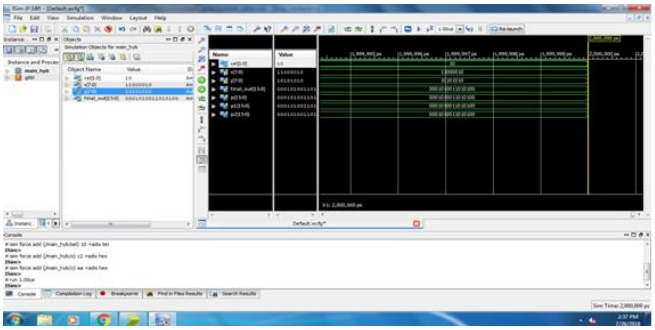


Fig.6. Simulation results.

In this recreation result, the sources of info are given and the mux will be chosen to choice line is "00".in this condition there is no yield why on the grounds that the mux will be doesn't perform in "00"condition, that is the reason the yield is zero.



To consider the above simulation results, the determination line is"01".the corner duplication is continued to ascertain the information operands to produce the yield, and afterward the mux will be chosen to the stall augmentation yield.



Considering above simulation results the selection line of mux is “10”.in this condition the hybrid adder output will be selected. then the mux output is hybrid adder output.

CONCLUSION

Presently, from the idea of the current outline adders plan we have proposed CSA,CLA based snake outline modules to execute combination MAC squares are:

i. Modified Booth

ii. Hybrid adder

iii. Wallace multiplier/adder

Directly we broke down and assessed the proposed plan modules as specified above utilizing verilog. The estimation of every viper is arranged and modules as charts appeared in results and exchanges. Every module are integrated and recreated utilizing Xilinx 14.2.the relative examination of proposed modules with "Cushion" and existing outline are being organized as demonstrated as follows.

S.NO	Parameters	Existing design	Proposed design
1	Area	15.7	9.7
2	Power	2.344W	0.97W
3	Delay	214us	96us

REFERENCES

[1]S. Goel, A. Kumar, M.A. Bayoumi, Design of robust, energy-efficient full adders for deep-submicrometer design using hybrid-CMOS logic style, IEEE Trans. Very Large Scale Integr. (VLSI) Syst, 14 (12) (2006), pp. 1309-1321, Record in Scopus

[2]P. Bhattacharyya, B. Kundu, S. Ghosh, V. Kumar, A. Dandapat, Performance analysis of a low-power high-speed hybrid 1-bit full adder circuit(2014)

[3]Z. Abid, H. El-Razouk, D. El-Dib, Low power multipliers based on new hybrid full adders Microelectr. J., 39 (12) (2008), pp. 1509-1515, ArticleDownload PDFView Record in Scopus

[4]S. Goel, M. Elgamel, M. Bayoumi, Y. Hanafy, Design methodologies for high- performance noise-tolerant XOR-XNOR circuits, IEEE Trans. Circ. Syst. I Reg. Papers, 53 (4) (2006), pp. 867-878, CrossRefView Record in Scopus

[5]K. Navi, M. Maeen, V. Foroutan, S. Timarchi, O. Kavehei A novel low-power full-adder cell for low voltageIntegration, 42 (4) (2009), pp. 457-467ArticleDownload PDFView Record in Scopus

[6]K. Navi, V. Foroutan, M. Rahimi Azghadi, M. Maeen, M. Ebrahimpour, M. Kaveh, et al.A novel low-power full-adder cell with new technique in designing logical gates based on static CMOS inverterMicroelectr. J., 40 (10) (2009), pp. 1441-1448Article Download PDFView Record in Scopus

[7] I. Brzozowski, A. KosDesigning of low-power data oriented addersMicroelectr. J., 45

(9) (2014), pp. 1177-1186Article Download PDFView Record in Scopus

[8]C.-K. Tung, Y.C. Hung, S.H. Shieh, G.S. HuangA low-power high-speed hybrid CMOS full adder for embedded system, Proceeding in Design and Diagnostics of Electronic Circuits and Systems(2007)

[9]H.T. Bui, Y. Wang, Y. JiangDesign and analysis of low-power 10-transistor full adders using novel XOR-XNOR gatesIEEE Trans. Circuits Syst. II Analog Digit. Signal Process, 49 (1) (2002), pp. 25-30 View Record in Scopus

[10] M. Alioto, G. Palumbo, Analysis and comparison on full adder block in submicron technology IEEE Trans. Very Large Scale Integr. (VLSI) Syst, 10 (6) (2002), pp. 806-823, CrossRefView Record in Scopus