A novel implementation for a 2^n-1 modular adder, using Carbon nanotube field effect transistors

Seyyed Ashkan Ebrahimi[1] and Mohammad Reza ReshadiNezhad[2]

[1,2]Department of Computer Engineering, University of Isfahan, Isfahan, Iran. Email: {ebrahimi, m.reshadinezhad}@eng.ui.ac.ir.

***ABSTRACT*:** The modulo 2^n±1 are the most used modulo set in residue number system (RNS). The modulo adder 2^n-1 are very important due to their variety applications in DSP, like addition/subtraction and multiplication in digital filters, cryptography, fault detection, fault correction and checksum. In this article, applying carbon nanotube transistors based on End Around Carry structure, a modulo adder 2^n-1 is implemented. The problems are analyzed and in order to fix them and improve its orbital parameters, a new design is proposed. By this design, the problem for two representations of zero are solved, and by taking advantage of a structure the fast calculation of carry, the orbital parameters are upgraded. The Circuits here are simulated by Hspice simulation tools and their performances are verified. Simulation results confirm the superior performance of this proposed design compared to the EAC (End Around Carry) structure.

I) INTRODUCTION

Addition is the most fundamental arithmetic operation [1] that can be implemented through more complex arithmetic circuits such as multiplication, division and exponentiation [2] [3], because, in these calculations, addition and subtraction circuits are widely used. Addition operations are vital components of the processors, due to their extensive application. In a logical computing unit, Floating Point calculations unit, and address calculation to access the cache and main memory [4]. From the beginning of digital systems and arithmetic circuits development, extensive studies have been conducted this issue which is still on going. With improving the performance of addition operations, orbital parameters of the derived circuits can be upgraded [5]. Hence, the implementation of this arithmetic operation in a fast and low-power way is a necessity to make a fast and low-power chip [6]. Thus the increased performance of the addition operations is one of the major challenges in the design and implementation of the arithmetic circuits. The main problem in implementing fast addition circuits is the carry propagation [1] [6]. The most significant bits are depended on least significant bits in order to calculate the addition, thus causing a significant increase in circuit delay [7]. Solutions are proposed to solve this problem, are: alternative numeral systems such as Redundant number system [1] and Residue number system [8]; and structures for fast carry propagation like Carry Look Ahead and Prefix adders [9]. Residue Number System (RNS) has a different mathematical approach to the numeral system. In RNS each weighted number is to be broken in to several smaller numbers so that the numbers are not weighted to each other. The arithmetic operations on these small numbers conducted in parallel [10] [11] [12]. With respect to delay, it can be assumed that in each processing only one number is processed. In RNS, with a conversion of large number (The number of bits) to several small numbers, a great calculating operation is divided into several small parallel calculations, so that each of these small calculations are carried out in a separate and independent manner [13] [14]. This feature shrinks the carry propagation chain. Useful in many applications to enhance computing speed. MOSFET transistors and CMOS technology have devoted the largest share on the implementation of digital circuits to themselves. Up to now many systems and tools such as computers, mobile phones, digital cameras, and other electronic tools are made through them. Major advances in this technology are due to the smaller size transistors. Scaling transistor will give the opportunity to increase the number of transistor inside a chip and thus enhance the functionality of the chip, which is pointed out by Moore's low [15]. Despite many advances in physics and chemistry field, continuation of this trend is faced with great challenges. So that, making the size of the transistors smaller, has its own serious constraints: Quantum effects and chip temperature rise. Investigating on alternative tools and technologies is inevitable. One of these tools is the Carbon Nanotube Field-effect Transistors (CNFETs) an achievement by nanotechnology with a great chance to replace the conventional silicon technology [16] [17] [18]. In these transistors the semiconductor carbon nanotubes are used as the transmission channel [19].

In the next two sections of this article Carbon Nanotube Transistor and Residue Number System respectively, are briefly reviewed. After RNS, the modulo adder 2^n-1 is examined. In the fourth section first, the EAC structures is implemented and after examining its advantages and disadvantages, the proposed structure would be introduced were carbon nanotube transistors is implemented. This section includes the detailed description, showing simulation results and introducing the proposal advantages. At the end summery and conclusion will be presented. The circuits are analyzed by Hspice simulation tools and also the Stanford library model [20] for CNFETs.

II) CARBON NANOTUBE TRANSISTORS

Carbon nanotube (CNT) is used as a transmission channel in CNFETs. This channel is placed between Source and Drain, near the Gate [15]. Carbon nanotubes can be considered as a sheet of graphite (Graphite is an allotrope of carbon) wrapped around itself forming a hollow cylinder. The number of carbon atom layers that form a hollow cylinder, can be more than one. Torsion direction of carbon atoms in length of atom will be displayed with a vector called Chirality (*ch*). For checking this vector, two unit vectors of a_1 and a_2 are used and the size of each vector is indicated by n_1 and n_2. The n_1 and n_2 are two positive integers as shown in Fig (1) [21]. According to the values n_1 and n_2, three different configurations for nanotube can be shaped, if one of these two vectors is zero, the nanotube will be Zigzag, If $n_1=n_2$, the nanotube is Armchair, otherwise, the nanotubes will be Chiral. In electrical conductivity, if $n_1=n_2$ or if $n_1-n_2=3i$ (i is an integer), the nanotube is conductive (metallic) and otherwise it is semiconductor.

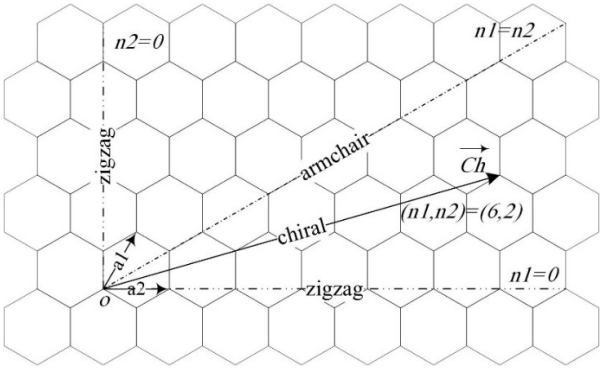

Fig.1 not radiated graphite sheet

Electrical resistance of nanotube is very low. Electrons are facing with little resistance, and can move along the nanotube. This amount of resistance is much lower than the prevalent conductors [21]. If the nanotube is composed of a single layer of carbon atom, called SWCNT and laminated called MWCNT [22]. To make CNFET transistors, single-walled semiconductor carbon nanotubes (SWCNT) are used. Like MOSFET, CNFETs have types P and N. The CNFETs advantage is that the size and mobility of N and P are identical [23]. Since the channels CNFETs are of a lower resistance than the MOSFETs, therefore the chip temperature and the power consumption can be reduced. The size of the *ch* vector is equal to the circumference of the base of carbon nanotubes obtained from Eq. (1) [24]. The diameter of the nanotube is obtained through dividing *ch* by π, which is an important parameter in circuit design. The threshold voltage of CNFET has inverse relation with its nanotube diameter. To calculate the threshold voltage of each CNT transistor with a diameter of *D* applying Eq. (2) [24]. So by changing the diameter of the nanotube, transistors with different threshold voltages can be produced. See CNFET, in fig (2).

$$V_{th} \approx \frac{E_{bg}}{2e} = \frac{\sqrt{3}}{3}\frac{aV_\pi}{eD_{CNT}} \approx \frac{0.436}{D_{CNT}(nm)} \quad (1)$$

$$D_{CNT} = \frac{a\sqrt{n_1^2+n_1n_2+n_2^2}}{\pi} \approx 0.0783\sqrt{n_1^2 + n_1n_2 + n_2^2} \quad (2)$$

Where, e is the electronic charge, E_{bg} is the Bandgap, a is the space of carbon atoms at approximately 0.249, D_{CNT} is the diameter of the nanotube, and the value of V_π is approximately 3.033.

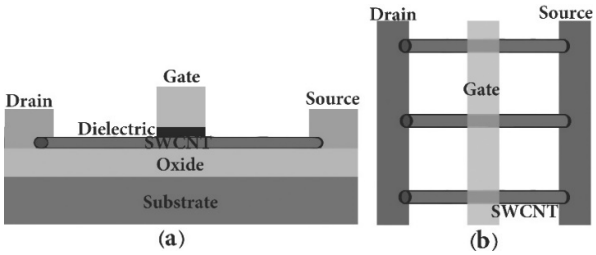

Fig.2 a) Front view and b) top view the of a CNFET

III) RESIDUE NUMBER SYSTEM

To convert the common numbers to Residue Number System they are broken into several smaller numbers based on a set called Moduli set. Moduli set is the basis of RNS. Each Moduli Set consists of a series of components called moduli. The number and choosing manner of the modules plays an important role in the system efficiency [8]. Selecting Moduli Set will have a direct effect on the orbital parameters and is determinant to evaluate overall system efficiency [14]. The moduli set are prime numbers in respect to each other. In general, the Moduli Set of *M* With m_i components (moduli) is shown as follows:

$$M=\{m_1, m_2, m_{3,\dots,} m_n\} \quad (3)$$

The Dynamic Range (DR) presents the set of number covered by RNS. In other words, DR determines which domain from the conventional number system selected by the moduli set as residue. This value is obtained by multiplying the modules together [13].

$$DR=m_1 \times m_2 \times \dots \times m_n \quad (4)$$

Each numbers in the range [0, DR), have a unique presentation in RNS. To calculate the residue corresponding number x than modulo m, desired number is divided by m and remainder of the division will be presentation of x for the modulo m. This sequence is presented by following two equations.

$$X \xrightarrow{RNS} (x_1, x_2, \dots, x_n) \quad (5)$$

$$x_i = X \bmod m_i = |X|_{m_i} \implies 0 \le x_i \le m_i \quad (6)$$

In a RNS system, the task of converting the conventional number into a residue number is through a unit called Forward Converter. Which converts the binary number to residue numbers [8]. Modulo presentation of number *x* in modulo *m* is shown in Eq. (7).

$$|x|_m = \begin{cases} x & if\ 0 \le x \le m-1 \\ x + km & if\ x < 0 \\ x - km & if\ x \ge m \end{cases} \quad (7)$$

Where *k* is a positive integer, *m* is the desired modulo and *x* is any arbitrary number. Required arithmetic operations are performed on each one of the residue individually. The circuits responsible for this calculation called are Modular Channels. One of the most important modulo channels is the Modular Adders. These adders are one of the most fundamental and primary elements used in residue number system [8]. The extents of these elements are such that they are used in all different parts of the RNS, such as forward converter, arithmetic units, and reverse converter. Implementing RNS circuits without modulo adders is practically impossible. The arithmetic operations for each of the numbers (residues) are carried out in parallel with other residues and by circuits designed to work in a particular modulo. After conduction calculations through Modulo Channels, the results of each modulo will be transmitted to a Reverse Converter [14] where they are, converted into a weighted number (conventional number) [8]. If $Y=\{y_1, y_2, ..., y_n\}$ and $X=\{x_1, x_2, ..., x_n\}$ are residues *X* and *Y* with respect to Moduli Set $M=\{m_1, m_2, ..., m_n\}$, Then $x \pm y$ is equal to:

$$\{x_1 \pm y_1, x_2 \pm y_2, \dots, x_n \pm y_n\} \quad (8)$$

$$X \pm Y \cong \langle |x_1 \pm y_1|_{m_1}, |x_2 \pm y_2|_{m_2} \dots\dots\dots |x_n \pm y_n|_{m_n} \rangle \quad (9)$$

The overall structure of a RNS system is illustrated in Fig (3).

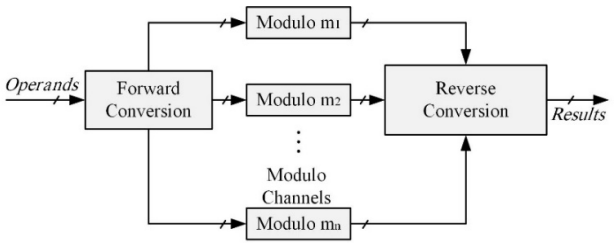

Fig.3 RNS System

III-I) THE MODULAR ADDER FOR MODULO 2^N-1

The modulo adder 2^n-1 which is equivalent to One's complement adder, in addition to using RNS has many other applications, like addition/subtraction and modulo multiplication in digital filters [25], Cryptography [26], error detection and correction [27], as well as calculation of checksum in the fast networks [28]. Various structures are presented for the modulo adder 2^n-1, [29] [30] [31] [32] [33] [34] each with specific characteristics. According to Eq. (7), addition in this modulo can be implemented based on Eq. (10).

$$|A + B|_{2^n-1} = \begin{cases} A + B - (2^n - 1) & if\ A + B \ge 2^n - 1 \\ A + B & if\ A + B < 2^n - 1 \end{cases} \quad (10)$$

Since $A+B-(2^n-1)=(A+B+1)-2^n$, then Eq. (10) can be written as:

$$|A + B|_{2^n-1} = \begin{cases} |A + B + 1|_{2^n} & if\ A + B \ge 2^n - 1 \\ A + B & if\ A + B < 2^n - 1 \end{cases} \quad (11)$$

Since *A* and *B* are in modulo 2^n-1 and their largest value is $(\underbrace{11\dots1}_{n})$-1, then the condition $A+B \ge 2^n-1$ will be true, when *A* and *B* are complement each other and or carry-out is 1. Thus Eq. (11) can be rewritten in this form:

$$|A + B|_{2^n-1} = \begin{cases} |A + B + 1|_{2^n} & if\ A + B \ge 2^n \\ A + B & otherwise \end{cases} = |A + B + Carry|_{2^n} \quad (12)$$

Eq. (12) is a general method known as End Around Carry (EAC). In this method, the carry-out must be added to the sum of two numbers. Therefor the final answer is found in two steps. In the first step, the carry-out is calculated by using a CPA, where, the value of carry-in to CPA considered zero. In the second step the carry-out obtained in the previous step will be applied to the CPA as the carry-in and add operation will repeated (fig 4). It should be noted that the adder is *n*-bit, hence the calculated sum will be *n*-bit (second carry of addition is ignored).

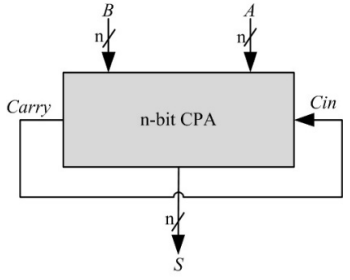

Fig.4 modulo 2^n-1 adder with EAC method

Delay and hardware cost of circuit in Fig (4) is:

$$\begin{cases} Delay = 2n(D_{FA}) \\ Area = n(A_{FA}) \end{cases} \quad (13)$$

In Eq. (13), the *DFA* is the full-adder delay and the *AFA* is the full-adder hardware cost.

IV) THE PROPOSED 2^N-1 ADDER

IV-I) IMPLEMENTATION OF EAC STRUCTURE

As described in the previous section, one way to design modulo 2^n-1 adders is binding carry-out of a CPA to its carry-in, known as EAC method (fig 4). The EAC structure for implementing VLSI requires an n-bit CPA. For this purpose a Ripple Carry Adder (RCA) is used. A schematic diagram of the RCA is shown in Fig (5). RCAs are the easiest and least costly form of CPA.

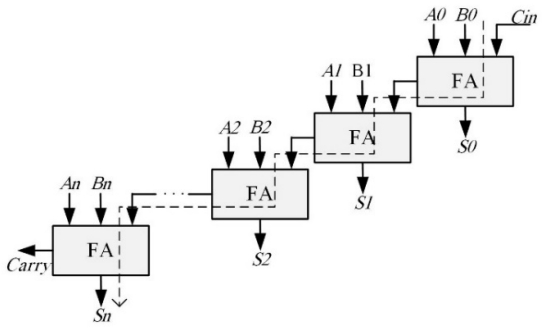

Fig.5 schematic diagram of the RCA

The dotted line in Fig (5) shows the critical path of the delay. RCA is made of cascade connection of several FAs. For this purpose, the conventional structure of FA (fig 6) which is implemented by carbon nanotubes transistors has been used.

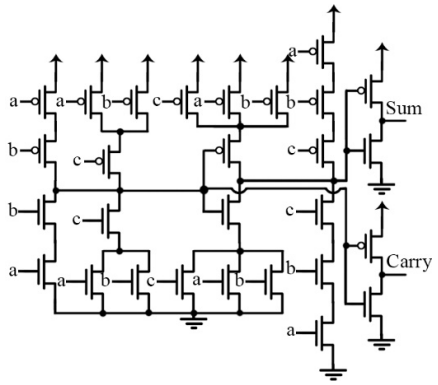

Fig.6 the conventional structure of FA

The most important advantage of the EAC is the simplicity of its design and implementation.

Furthermore, it is possible that any kind of CPA be used in its structure. Also, expansion of *n* is easily done due to its simple structure.

IV-II) PROPOSED DESIGN

EAC scheme has three drawbacks: 1) is greatly delayed, because the ultimate answer is prepared in 2 general cycles, carry-out calculation and calculate the sum with carry-in value that is calculated in the previous step. 2) Binding the carry-out to carry-in will cause a conditional loop, 3) this style of design led to two representation of zero. For example, if $n=4$ and numbers $A=(1000)_2=(8)_{10}$ and $B=(0111)_2=(7)_{10}$ with moduli set $15\ (2^4-1=15)$. Since the carry-out of these two numbers (assuming that the carry-in is '0') is zero. The sum output of EAC, will be $S=(1111)_2=(15)_{10}$. But it is expected that, according to Eq. (7), since the sum is equal to moduli, then the value of the moduli will be subtracted from *S* and final result is $(0000)_2$. This example indicate that in EAC structure there are two representation of zero, one of them negative zero $(111\ldots1)_2$ and the other positive zero $(000\ldots0)_2$. This can be trivial in some applications, such as reverse converter design in a residue number system. While, these circuits are used in the RNS, their correction in single representation of zero is not required [34]. In most applications of this arithmetic circuit, the two zero representation can be problematic. In general, adders that have only single representation of zero, have additional hardware overhead.

Assuming two single bit numbers *a, b* we have:

$$p = a \oplus b, g = ab \quad (14)$$

Where *p*=propagate that is, the carry propagation and *g*=generate that is, the carry generation and $\oplus$ denotes the XOR. The carry-out of these two single-bit numbers can be obtained from the following:

$$Carry = g + pC_{in} \quad (15)$$

For two four-bit numbers Eq. (15) can be written as follows:

$$Carry = g_3 + p_3g_2 + p_3p_2g_1 + p_3p_2p_1g_0 + p_3p_2p_1p_0C_{in} \quad (16)$$

In modulo adder through EAC method the carry-in is always considered as being '0'; thus Eq. (16) can be summarized as follows:

$$Carry = g_3 + p_3g_2 + p_3p_2g_1 + p_3p_2p_1g_0 \quad (17)$$

In implementing of the modulo adder 2^n-1 through EAC method the problem of two representation of zero is retrieved when both the inputs are complementary. In Eq.

(18), if *x='1'* the problem of two representation of zero is obvious and *(000 ... 0)* is displayed as *(111 ... 1)*.

$$x = (a_1 \oplus b_1)(a_2 \oplus b_2) \dots (a_n \oplus b_n) \qquad (18)$$

In this proposal, Eq. (17) is used in order to calculate the carry-out and Eq. (18) is used in order to determine the condition that causes two representation of zero. For hardware implementation these two equations are merged. Schematic diagram of the proposed scheme is shown in Fig (7). This proposal consists of two separate units: a) the carry computation unit (Carry Generator: CG) and b) the n-bit Carry Propagation Adder (CPA). CG unit contains the detecting circuit of two representation of zero condition and calculation of carry-out. These two circuits are combined with each other. The dotted line in Fig (7) shows the critical path of delay in this proposed. In addition to CPA a separate circuit is applied to calculate the carry, thus the loop problem in EAC structure is removed.

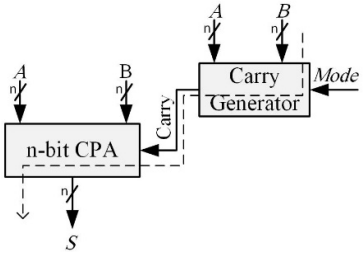

Fig.7 Schematic diagram of the proposed adder

As observed in Fig (7), the carry is calculated through the CG unit and as well as the condition of double representation of zero. This circuit is designed to calculate the carry in a rapid manner. The output of the CG unit enters to *n*-bit CPA unit. In a sense that it is considered as a carry-in of CPA.

As mentioned earlier, in order to build CG unit, Eq. (17 & 18) have been implemented. The field effect transistor structure of this unit is shown in Fig (8). Here, all the transistors simultaneously initialized according to the given inputs and the ON and OFF status of each transistor is determined. The proposed method can perform in single zero format or double zero format by using a signal named "Mode". The signal is used in converting the zero formats vice versa. Since none of the transistors in this structure anticipate carry, therefor carry-out is calculated in a fast manner. The hardware overhead imposed on the circuit for removing the problem of two representation of zero is two transistors.

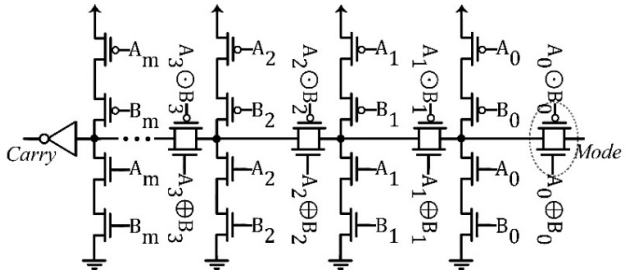

Fig.8 Transistor level structure of carry generator unit

If the value of Mode is '1', the adder will operate as two representation of zero and if the initial value of the Mode is '0', the addition operation is conducted in a manner that there would be a single representation of zero. The imposed overhead for controlling the single or two representation of zero is illustrated in Fig (8).

For the b section of the circuit (*n-bit* CPA), the RCA introduced is used [35]. One advantage of this proposal is the existence of signals $p_i, \overline{p_i}$ (Fig 10); these signals are used in controlling the (MUXs). Consequently, for producing $(a_i \oplus b_i) and\ (a_i \odot b_i)$ in CG unit, there is no need for extra transistors and the generated signals in CPA unit can be used. Another advantages of introduced proposal in [35], is using the style of bridge design in implementing minority-not gate. This style of design enhances the efficiency of the circuit where the delay and power consumption are reduced [36]. MUXs of Adder circuit in [35] are designed through the Pass gate technology; therefore, the outputs are Full-swing.

The last FA in CPA structure (unit B) is without carry calculation section. Since the sum of the two *n*-bit numbers in modulo 2^n-1 are of *n* bits.

The transistor level structure of CPA unit for *n=4* is shown in Fig (9).

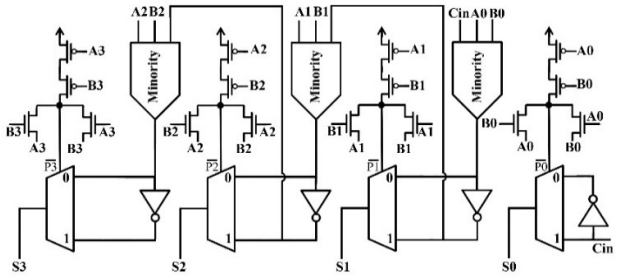

Fig.9 Transistor level structure of CPA unit

As observed in Fig (8), the CG circuit is implemented with the use of Pass-gate technology. Thus, with increasing the number of bits (*n*), the Fan-out of the circuit increases casing in increase delay. To remove this problem the CG unit is divided into blocks of four bits, that the output of the *(k-1)*-

th block, after passing through an inverter is connected to the input Mode of the *k*-th block (Fig 10).

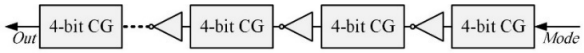

Fig.10 Schematic diagram of CG unit for *n* larger than 4

The circuits presented in this article are implemented through carbon nanotube field effect transistors. Hspice simulation tool and the Model in [20] are used for circuits simulation. The model is fully described in the two articles [37] [38]. The number of nanotube per transistor is *3*, the value of *pitch=30e-9m* and gate *length=32e-9m.* To calculate the delay in the circuit, all changes in input and output are evaluated and the maximum delay is reported as the circuit delay. Parameter PDP is obtained by multiplying average power consumption in the worst-case delay. Obtaining the values of delay and power consumption, as well as analyzing the input and output pulses and verifying the circuit performance, the ComsmosScope software is used. The nanotubes diameter used in this proposals with corresponding Chirality and each threshold voltage is shown in Table (1).

Table.1 Diameter, Chirality and threshold voltage of CNFETs that used in this article

Type	Diameter	Chirality	Threshold
P-Type	1.487	(19,0)	-0.289
N-Type	1.487	(19,0)	0.289

The simulation results at 27 °C and supply voltage of 1.2 *V* and without capacitive load for *n={16, 8, 4}* are listed in Table (2). The input pattern applied to the circuits for *n=4* is shown in Fig (11).

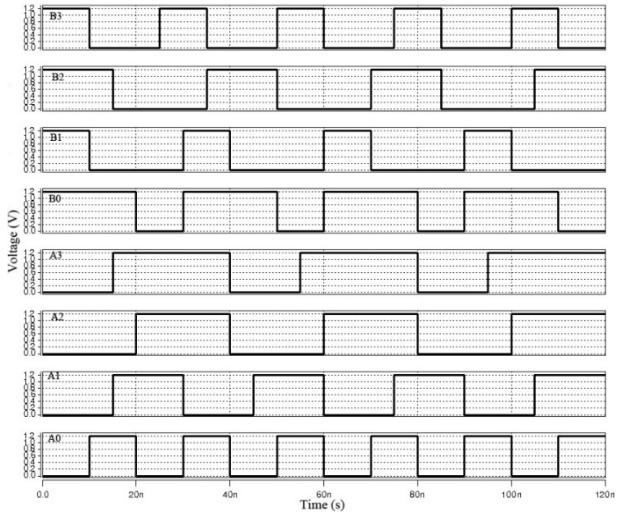

Fig (11) Inputs pattern for *n=4*

Table (2) Simulation results of the proposed adder

n	Design	Delay(ps)	Power(uW)	PDP(e-17)	Tr. Count
4	1	28.277	0.2596	0.7341	112
	2	17.696	0.2465	0.4362	104
8	1	55.978	0.7991	4.4734	224
	2	40.908	0.3718	1.5209	220
16	1	109.265	4.5031	49.203	448
	2	89.459	1.9534	17.4749	452

V) CONCLUSION

As far as the authors have are concerned in this article for the first time the modulo adder 2^n-1 by using Carbon nanotube transistors is implemented. First, one adder based on EAC structure is implemented and evaluated, then a proposal to remove the problems of EAC structure and enhance its efficiency is presented. This introduced proposal, by using CG structure with the ability to compute the carry-out in a fast manner, upgrades the delay parameter properly. Here the problem of two representation of zero is solved, in a manner that the introduced circuit is able to calculate as single or two representations of zero. The structure is designed in a manner that does it not impose a large hardware overhead. Power consumption is significantly reduced by using the style of bridge design and the Pass-gate. Simulation results confirm the superior performance of this proposal compared to that of the EAC structure.

REFERENCES

[1] B. Parhami, "Computer arithmetic Algorithms and Hardware Designs," New York, Oxford university press, 2000.

[2] N. Weste and K. Eshraghian, Principles of CMOS VLSI Design: A Systems Perspective, Addision-Wesley, 1993.

[3] R. Zimmermann and W. Fichtner, "Low-power Logic Styles: CMOS versus Pass-transistor Logic," *IEEE Journal of Solid-state Circuits*, vol. 32, pp. 1079-1090, 1997.

[4] J.-G. Chung, J. Yingtao, A.-S. Abdulkarim, W. Yuke and S. Edwin, "A Novel Multiplexer-Based Low-Power Full Adder," *IEEE Transactions on Circuits & Systems*, vol. 51, no. 7, pp. 345-348, July 2004.

[5] K. Navi, H. H. Sajedi, R. F. Mirzaee, M. H. Moaiyeri, A. Jalali and O. Kavehei, "High-speed full adder based on minority function and bridge style for nanoscale," *Integration, the VLSI journal*, vol. 44, no. 3, pp. 155-162, 2011.

[6] I. Koren, "Computer Arithmetic Algorithms," in *Computer Arithmetic Algorithms*, A K Peters, 2002, pp. 93-138.

[7] M. Lu, "Arithmetic and Logic in Computer Systems," in *Arithmetic and Logic in Computer Systems*, Wiley Intercience, 2004, pp. 29-52.

[8] A. Omondi and B. Premkumar, "Residue Number Systems: Theory and Implementation," in *Residue Number Systems: Theory and Implementation*, London, Imperial College Press, 2007.

[9] R. Zimmermann, *Computer Arithmetic: Principles, Architectures, and VLSI Design*, Z¨urich: Integrated Systems Laboratory, ETH Z¨urich, 1998.

[10] M. R. Reshadinezhad and F. Samani Kabiri, "A Novel Low Complexity Combinational RNS Multiplier Using Parallel Prefix Adder," *IJCSI International Journal of Computer Science Issues,* vol. 10, no. 3, pp. 430-440, 2013.

[11] M. A. Bayoumi and P. Srinivasan, "Parallel arithmetic: from algebra to architecture," in *Proceding IEEE Symposium on Circuits and Systems,* 1990.

[12] T. Stouratitis and V. Paliouras, "Considering the alternative in low powerdesign," *IEEE Circuit and Devices,* vol. 7, pp. 23-29, 2001.

[13] K. Navi, A. Sabbagh Molahosseini and M. Esmaeildoust, "How to Teach Residue Number System to Computer Scientists and Engineers," *IEEE transactions on education,* vol. 54, no. 1, pp. 156-163, 2011.

[14] A. Sabbagh Molahosseini, k. Navi, C. Dadkhah, O. Kavehei and S. Timarchi, "Efficient Reverse Converter Designs for the New 4-Moduli Set Based on New CRTs," *IEEE transactions on circuits and systems—I,* vol. 57, no. 4, pp. 823-835, 2010.

[15] "Bio-inspired and nanoscale integrated computing," Hoboken, New Jersey, John Wiley & Sons, Inc, 2009.

[16] M. R. Reshadinezhad, M. H. Moaiyeri and K. Navi, "An Energy-Efficient Full Adder Cell Using CNFET Technology," *IEICE Transactions,* vol. 95, no. 4, pp. 744-751, 2012.

[17] A. Raychowdhury and K. Roy, "Carbon-nanotube-based voltage-mode multiple-valued logic design," *IEEE Transactions on Nanotechnology,* vol. 4, no. 2, pp. 168-179, 2005.

[18] S. Lin, Y.-B. Kim and F. Lombardi, "CNTFET-Based Design of Ternary Logic Gates and Arithmetic Circuits," *IEEE Transaction on nanotechnology,* vol. 10, no. 2, pp. 217-225, 2011.

[19] S. J. Tans, A. R. M. Verschueren and C. Dekker, "Room-temperature transistor based on a single carbon nanotube," *Nature,* vol. 393, pp. 49-52, 1998.

[20] S. University, "CNFET model Website," Stanford, 21 6 2012. [Online]. Available: http://nano.stanford.edu/model.php?id=23. [Accessed 12 1 2013].

[21] J. Deng, "Device Modeling And Circuit Performance Evaluation For Nanoscale Devices: Silicon Technology Beyond 45 nm Node And Carbon Nanotube Field Effect Transistors," Stanford university, 2007.

[22] P. Keshavarzian and K. Navi, "Universal ternary logic circuit design through carbon nanotube technology," *International Journal of Nanotechnology,* vol. 6, no. 10, pp. 942-953, 2009.

[23] G. Cho, Y. B. Kim, F. Lombardi and M. Choi, "Performance evaluation of CNFET-based logic gates," *Instrumentation and Measurement Technology Conference, 2009. I2MTC '09. IEEE,* pp. 909 - 912, 2009.

[24] M. H. Moaiyeri, K. Navi and O. Hashemipour, "Design and Evaluation of CNFET-Based Quaternary Circuits," *Circuits, Systems, and Signal Processing,* vol. 31, no. 5, pp. 1631-1652, 2012.

[25] W. Jenkins and B. Leon, "The Use of Residue Number," *IEEE Transaction on Circuits and Systems,* vol. 24, pp. 171-201, 1977.

[26] X. Lai and J. Massey, "A proposal for a new blcok encryption," in *in Advances in Cryptology - EUROCRYPT'90,* Berlin, Germany, 1990.

[27] S. S.-S. Yau and Y.-C. Liu, "Error Correction in Redundant," *IEEE Transaction Computers,* vol. 22, no. 1, pp. 5-11, 1984.

[28] F. Halsall, Data Communications, Computer Networks and Open Systems, Addison Wesley, 1996.

[29] Jun Chen and J. E. Stine, "Parallel Prefix Ling Structures for Modulo 2^n-1 Addition," *Application-specific Systems, Architectures and Processors, 2009. ASAP 2009. 20th IEEE International Conference on,* pp. 16 - 23, 2009.

[30] L. Kalampoukas, D. Nikolos, C. Efstathiou, H. T. Vergos and J. Kalamatianos, "High-speed parallel-prefix module 2n-1 adders," *Computers, IEEE Transactions on,* vol. 49, no. 7, pp. 673 - 680, 2000.

[31] R. A. Patel, M. Benaissa and S. Boussakta, "Efficient new approach for modulo 2n-1 addition in RNS," *Computers and Digital Techniques, IEE Proceedings,* vol. 153, no. 6, pp. 399 - 405, 2006.

[32] S. Timarchi, M. Fazlali and S. D. Cotofana, "A unified addition structure for moduli set {2n−1, 2n, 2n+1} based on a novel RNS representation," *Computer Design (ICCD), 2010 IEEE International Conference on,* pp. 247 - 252, 2010.

[33] B. Shaoqiang, W. J. Gross, Wei Wang, A. Al-Khalili and M. N. S. Swamy, "An area-reduced scheme for modulo 2n-1 addition/subtraction," *System-on-Chip for Real-Time Applications, 2005. Proceedings. Fifth International Workshop on,* pp. 396 - 399, 2005.

[34] R. A. Patel, M. Benaissa and S. Boussakta, "Fast Parallel-Prefix Architectures for Modulo 2n-1 Addition with a Single Representation of Zero," *IEEE Transactions on Computers,* vol. 56, no. 11, pp. 1484-1492, 2007.

[35] S. A. Ebrahimi and P. Keshavarzian, "Fast low-power full-adders based on bridge style minority function and multiplexer for nanoscale," *International Journal of Electronics,* vol. 100, no. 6, pp. 727-745, 2013.

[36] K. Navi, M. H. Moaiyeri, R. Faghih Mirzaee, O. Hashemipour and B. Mazloom Nezhad, "Two new low-power Full Adders based on majority-not gates," *Microelectronics Journal,* vol. 40, pp. 126-130, 2009.

[37] J. Deng and H. S. P. Wong, "A compact SPICE model for carbon-nanotube field-effect transistors including nonidealities and its application—Part I: Model of the intrinsic channel region," *IEEE Transaction on Electron Device,* vol. 54, no. 12, pp. 3186-3194, 2007.

[38] J. Deng and H. S. P. Wong, "A compact SPICE model for carbon-nanotube field-effect transistors including nonidealities and its application—Part II: Full device model and circuit performance benchmarking," *IEEE Transaction on Electron Device,* vol. 54, no. 12, pp. 3195-3205, 2007.