International Journal of Research
Available at https://edupediapublications.org/journals
P-ISSN: 2348-6848
E-ISSN: 2348-795X
Volume 07 Issue05May 2020

Low Power VLSI Architecture for Modular Adder by Reversible Gates

Poojarieswaramma[1], nandini[2] (ASSISTANT PROFESSOR.)

Golden valley integrated campus, Kadiri road, Angallu post, Madanapalli, Chittoor, and Andhra Pradesh 517325.

ABSTRACT

Reversible logic is a computing paradigm that has attracted significant attention in recent years due to its properties that lead to ultra-low power and reliable circuits. Reversible circuits are fundamental, for quantum computing. Since addition is a basic operation, designing efficient adders is a cornerstone in the research of reversible circuits. Residue Number Systems (RNS) has been as a powerful to provide parallel and multiplications are dominant. In this paper, for the first time in the literature, we propose the combination of RNS and reversible logic. The parallelism of RNS is leveraged to increase the performance of reversible computational circuits. Being the most fundamental part in any RNS, in this work we propose the implementation of modular adders, namely modulo 2n-1 adders, using reversible logic. Analysis and comparison with traditional logic show that modulo adders can be designed using reversible gates with minimum overhead in comparison to regular adders.

Keywords— Residue Number System (RNS), Reversible Circuits, Modular Adder, Parallel-Prefix Adders.

I. INTRODUCTION

Researchers in academia and industry believe that Moore's law is ending, and even newly delivered deep-submicron transistors are not significantly more efficient than their previous generations. Therefore, new computing paradigms should be investigated in order to overcome the predicted performance wall which will be reached in 2020. This rebooting of computing has to be based on novel methods at different computing levels of design abstraction, including arithmetic and circuit level, in order to address the challenges of the emerging applications such as deep convolutional neural network (DNN) and internet-of-things (IoT).Residue Number System (RNS) is one unconventional number system that can provide fast and low-power implementation of additions and multiplications. RNS is a different approach of dealing and representing numbers that provide parallelism at arithmetic level.

This number system has been applied to efficient implementations for asymmetric cryptographic and digital signal processing (DSP). RNS is used nowadays to achieve also energy-efficient and high-performance implementation emerging applications, such as deep neural networks, communication networks and cloud storage. However, current implementations of RNS systems, on ASICs and FPGAs, are based on the CMOS technology, which is reaching its limit. Alternative methods and technologies, such as nano electronic, are considered to be used. One of these alternatives is Reversible Computing (RC), which can provide ultra-low power computational circuits.

In this paper, we propose the joint usage of approaches, Residue Number System and Reversible Computing, to achieve ultra-efficient computing paradigm for the emerging applications. The ability of RNS to perform highly parallel and carry-free

International Journal of Research
Available at https://edupediapublications.org/journals
P-ISSN: 2348-6848
E-ISSN: 2348-795X
Volume 07 Issue05May 2020

arithmetic is well suited for taking advantage of the features of reversible circuits. In other words, reversible logic can be efficiently used to implement RNS circuits. However, since all the available RNS structures are designed for ASIC implementation, rethinking of RNS architectures should be performed to adapt them to the properties of reversible circuits.

The fundamental part of RNS systems is modular addition, since all parts of RNS including forward and reverse conversion are based on modular additions. Hence, the first step to implement RNS systems based on reversible circuits requires the design of efficient modular adders using reversible logic gates. This paper presents the first implementation of modulo 2*n*-1 adders based on reversible gates. For these modular adders, which are frequently used in RNS structures; parallel-prefix and ripple-carry architectures are considered.

II. RESIDUE NUMBER SYSTEM ARCHITECTURE

The first step to architect a RNS is to select moduli set according to the target application constraints and requirements. The moduli set consists of pair-wise relatively prime numbers {*m*1, m2… *mn*}, being the dynamic range the sequence of integers that can be uniquely represented in RNS, i.e. [0, *M*-1] with M=*m*1×*m*2×…×*mn*. In order to decrease the complexity of hardware realization of RNS-based arithmetic, usually near power-of-two moduli are adopted, such as 2*n*-1, 2*n* and 2*n*+1. Among these moduli, the simplest one to deal with is the 2*n*, which does not require any specific modular arithmetic, jut the circuits for binary arithmetic. Apart from that, the most frequent co-prime number in moduli sets for RNS is 2*n*-1, since moduli 2*n*+1 is more complex and its representation requires on additional bit. Typical RNS moduli sets are {2*n*-1, 2*n*+*k*, 2*n*+1}, {2*n*-1, 2*n*, 2*n*+1-1}, {2*n*-1, 2*n*, 2*n*+1, 22*n*+1-1}, {2*k*, 2*n*-1,2*n*+1, 2*n*+1-1} and {2*n*-1, 2*n*, 2*n*+1, 2*n*+1-1, 2*n*-1-1}. The main arithmetic blocks of RNS are the forward converter, the modular arithmetic in the channels, and the reverse converter. The forward converter translates the weighted binary number (*X*) to the residues (*xi*'s), according to the moduli, as:

$$X \xrightarrow{Forward\ Conversion} (x_1, x_2, \ldots, x_n) \quad (1)$$

Where

$$x_i = X \bmod m_i = |X|_{m_i} \quad for\ i = 1 \ldots n \quad (2)$$

Note that *mod* indicates the remainder of de integer division of *X* by *mi*. Then, considering two numbers *A* and *B* as follows:

$$A = (a_1, a_2, \ldots, a_n) \quad (3)$$

$$B = (b_1, b_2, \ldots, b_n) \quad (4)$$

Modulo arithmetic operations can be performed on residues as follows:

$$S = A \bullet B = (s_1, s_2, \ldots, s_n) \quad (5)$$

where

$$s_i = |a_i \bullet b_i|_{m_i}, \bullet \in \{+, -, \times\} \quad (6)$$

Finally, a reverse converter maps the results in the RNS domain to the regular weighted representation, by using, for example, the Chinese remainder theorem (CRT). Other RNS operations such as sign detection, magnitude comparison and overflow handling are optional, according to the target application, and harder to perform in the RNS domain. It should be mentioned that general division cannot directly be performed in RNS, but division by a constant, one of the moduli of the set, i.e. scaling, is easier to perform.

A general structure of a typical RNS processor is shown in Figure 1. The RNS represented data is processed in parallel with no dependence or carry propagation between the processing units. The process of encoding the input data into RNS representation is called *Forward Conversion*, and the process of converting back the output data from RNS to

International Journal of Research
Available at https://edupediapublications.org/journals
P-ISSN: 2348-6848
E-ISSN: 2348-795X
Volume 07 Issue05May 2020

conventional representation is called *Reverse Conversion*.

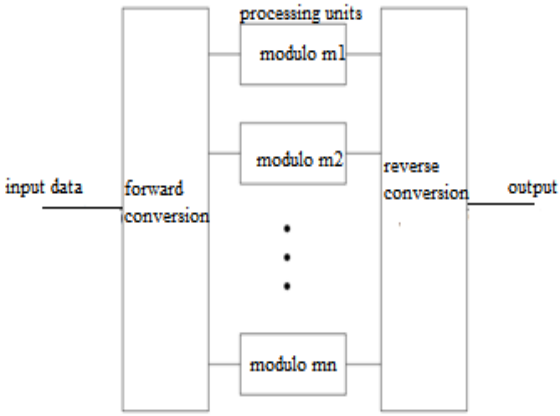

Fig1. General structure of an RNS-based processor

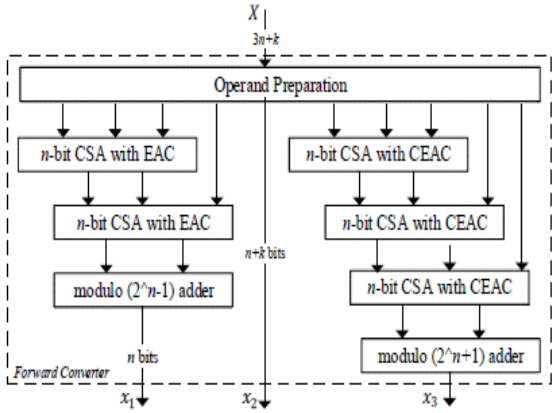

Fig.2. The forward converter for the moduli set{2*n*-1, 2*n*+*k*, 2*n*+1}

Most of the mentioned RNS operations are implemented using 3-to-2 carry-save adders (CSAs) with end-around carries (EACs) and 2-to-1 modular adders. A full hardware design of RNS with moduli set {2*n*-1, 2*n*+*k*, 2*n*+1} is reported in, and herein forward and reverse converters for this moduli set are depicted in Figs. 2 and 3, respectively. It can be observed that CSAs and carry-propagate modulo 2*n*-1 adders are the components required to implement a full RNS architecture, since arithmetic in a channel also requires modulo adders and multipliers. Thus, to have an efficient modular adder is fundamental for RNS-based applications.

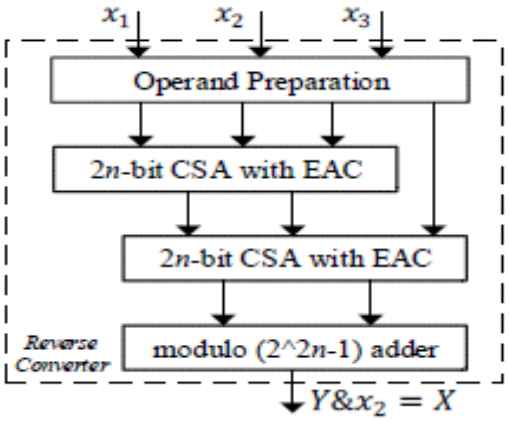

Fig 3. The full reverse converter for the moduli set {2*n*-1, 2*n*+*k*, 2*n*+1}

The CSA with EAC consists of independent full adders (FAs) which just combine the three inputs into two carry-save output vectors, as shown in Fig. 4. Its delay is just the delay of a single FA, while the overall area linearly depends on the width of the operands.

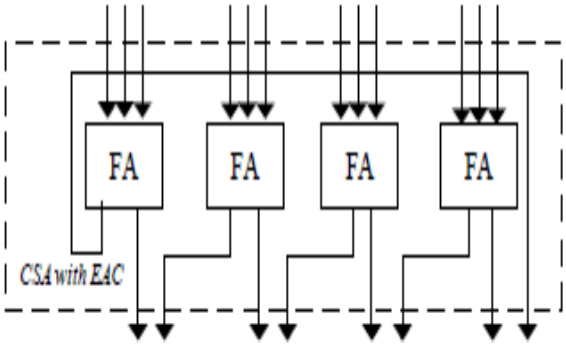

Fig. 4. The 4-bit CSA with EAC structure.

Note that CSA with CEAC is similar to CSA with EAC; just the end-around carry is complemented. Modular carry propagate adders can be designed based on different architectures, from low-cost ripple-carry adders (RCAs) (Fig. 5) to fast parallel-prefix adders (PPAs). The PPAs architectures can provide a good trade-off between circuit's parameters, being popular in RNS arithmetic circuits.

International Journal of Research
Available at https://edupediapublications.org/journals
P-ISSN: 2348-6848
E-ISSN: 2348-795X
Volume 07 Issue05May 2020

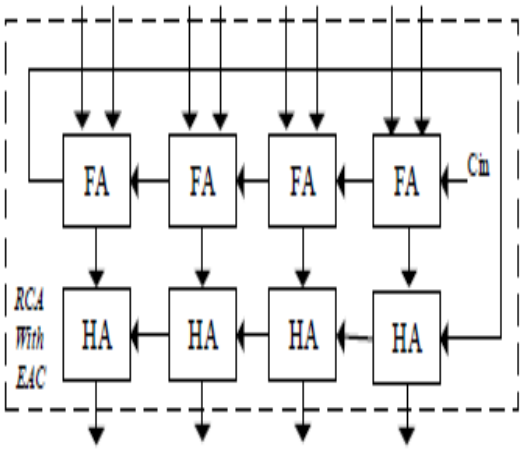

Fig 5. The 4-bit modulo 2*n*-1 adder based on ripple-carry method, namely RCA with EAC.

KOGGE STONE ADDER

KSA is a parallel prefix form carry look ahead adder. It generates carry in O (logn) time and is widely considered as the fastest adder and is widely used in the industry for high performance arithmetic circuits. In KSA, carries are computed fast by computing them in parallel at the costof increased area. The complete functioning of KSA can be easily comprehended by analyzing It in terms of three distinct parts .The working of KSA can be understood by the following Fig. 1 which corresponds to 4-bit KSA..

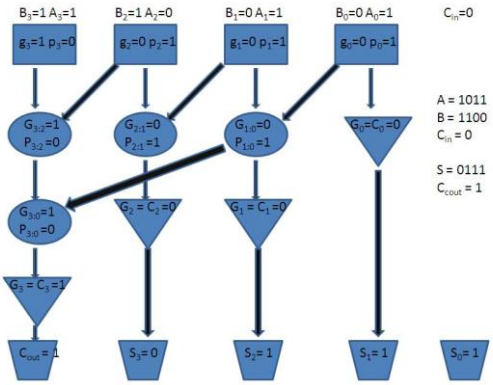

Fig. 6 Illustration of 4 bit KSA

III. MODULAR ADDER DESIGN USING REVERSIBLE CIRCUITS

(A)REVERSIBLE GATES

Reversible circuits provide a one-to-one relation between inputs and outputs, therefore inputs can be recovered from outputs. This interesting feature results in significant power saving in digital circuits. Classical digital gates are not reversible, reversible gates should be designed as basic components to design logical reversible circuits. Well known reversible gates are Feynman, Peres and HNG. The Feynman or controlled not (CNOT) gate is frequently used in reversible circuits, since it can provide exclusive OR (XOR) as well as copy and complement of the input. Since reversible circuits do not take advantage of fan-out, this gate can be used to achieve two copies of the same input by setting the other input of the gate to the zero-logic level. Similarly, by setting the second input of the CNOT to one-logic level, we can achieve the complement of the other input. A reversible logic gate has the same number of inputs and outputs with one-to-one correspondence between the input and the output. Hybrid new gate (HNG) is a universal reversible gate and realizes all Boolean functions.

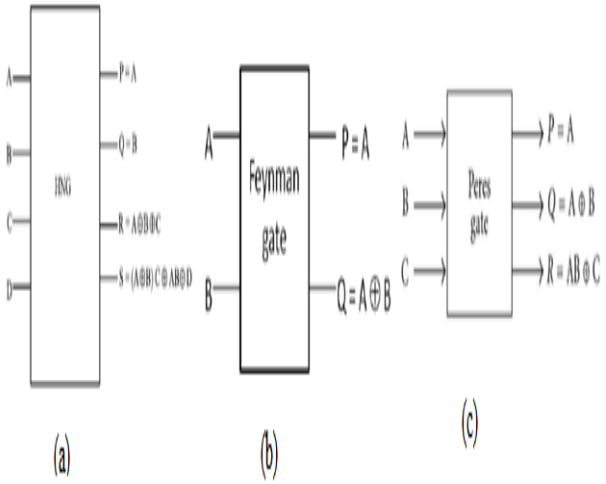

Fig 7: (a) HNG gate, (b) Feynman gate and (c) Peres gate

(B)The CSA with EAC

International Journal of Research
Available at https://edupediapublications.org/journals
P-ISSN: 2348-6848
E-ISSN: 2348-795X
Volume 07 Issue05May 2020

The CSA is a 3-to-2 compression unit that is very popular for regular arithmetic as well as in RNS architectures due to its speed and cost. According to Fig. 4, a CSA can be built by using *n* FAs for adding three *n*-bit operands. According to, the HNG reversible gate can be used to realize a FA by setting the fourth input of HNG to the zero-logic level, as shown in Fig. 6.

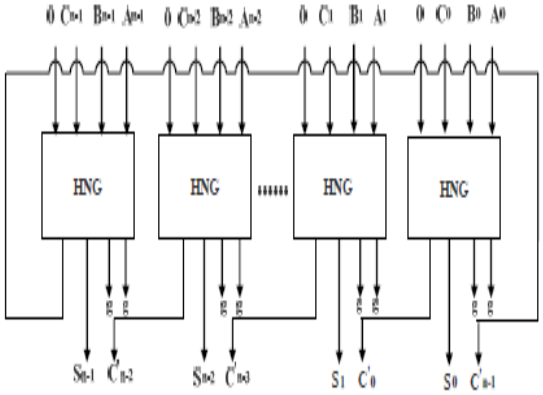

Fig. 8. The CSA with EAC using HNG reversible gates.

The quantum depth and cost of a HNG gate is 5Δ and 6, respectively. Therefore, the total quantum depth and cost of a *n*-bit CSA with EAC will be 5Δ and 6*n*, respectively, since the delay of a CSA is equal to the delay of just one FA. Besides, the final reversible circuits will have *n* constant inputs and 2*n* garbage outputs.

(C)The RCA-based Modulo Adder

As shown in Fig. 4, the RCA with EAC for modulo 2*n*-1 addition of two *n*-bit numbers, requires *n* FAs and *n* HAs in the first and second levels, respectively. Similar to CSA, FAs can be realized with HNG gates. Besides, the Peres reversible gate can be used to implement a HA, where the third input bit is set to zero, as shown in Fig. 9. The final quantum cost of the RCA with EAC for two *n*-bit operands is 6*n*+4*n*=10*n*, since the individual quantum cost and depth of a Peres gate is 4. Besides, the total quantum depth of the RCA with EAC is ((3×(*n*-1)+4+(3×(*n*- 1)+5))Δ. Furthermore, the total constant inputs and garbage outputs are 2*n* and 3*n*, respectively, since one of the inputs of HNG and Peres gates is zero, and also two and one outputs of HNG and Peres gates, respectively, are not used.

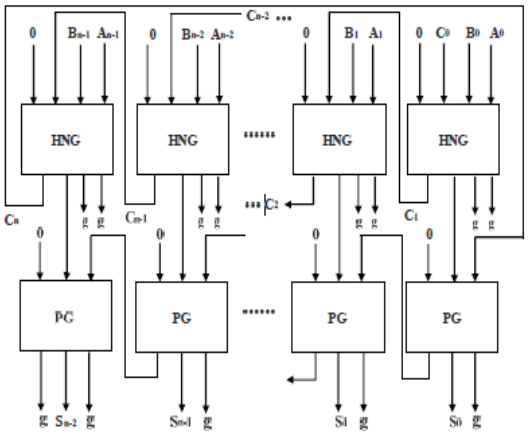

Fig. 9. The RCA with EAC using HNG and Peres reversible gates.

(D)Brent-Kung Adder

Brent-kung adder is one of the parallel pre-fix adders used for speeding of the operation. Parallel prefix adders are the fast adders and perform the arithmetic operations at the fast rate. Most Industries are using parallel prefix adders because of their advantages compare to other adders. Parallel prefix adders are faster and area efficient. Parallel prefix adder is a technique for increasing the speed in DSP processor while performing addition.

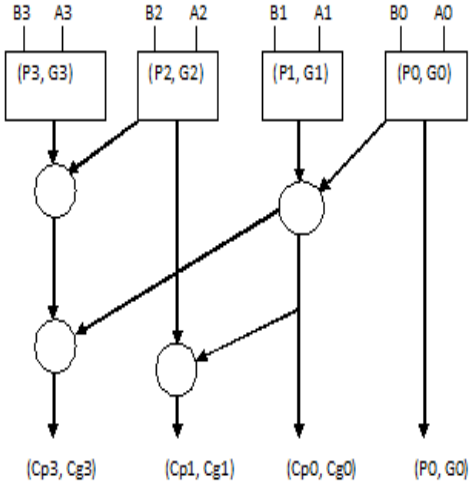

Fig10: 4-bit Brent Kung adder

International Journal of Research
Available at https://edupediapublications.org/journals
P-ISSN: 2348-6848
E-ISSN: 2348-795X
Volume 07 Issue05May 2020

IV. RESULTS

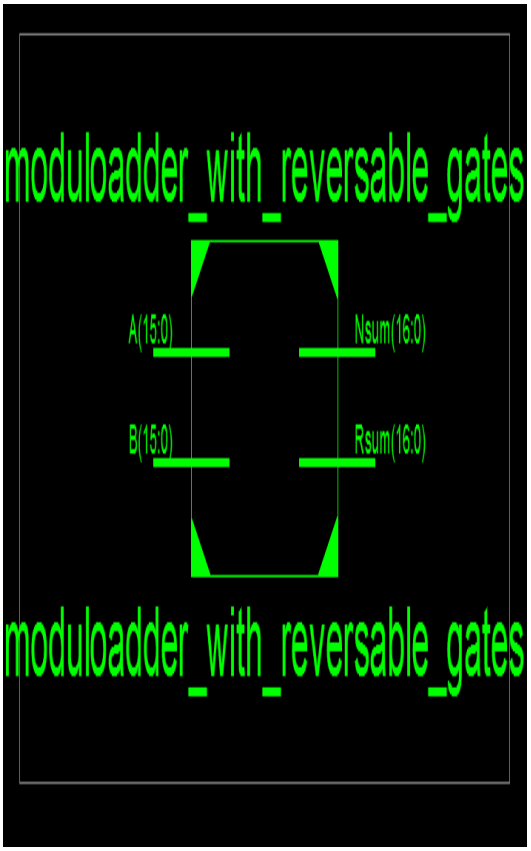

Fig11: RTL Schematic

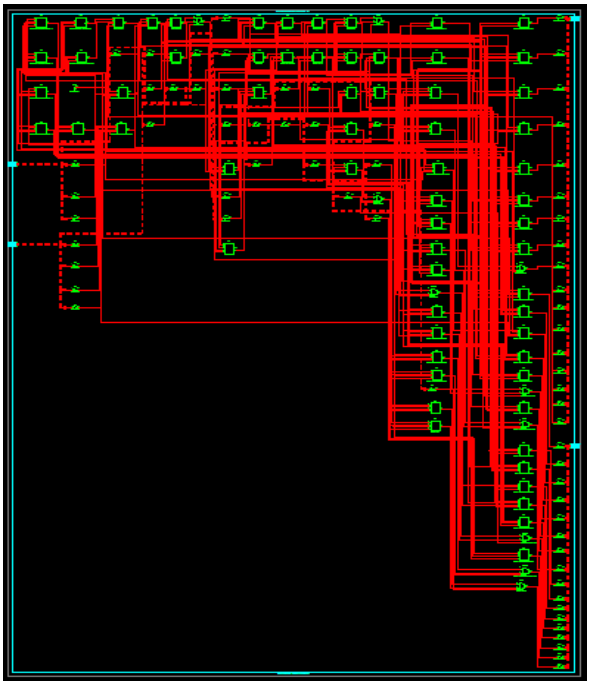

Fig12: View technology schematic

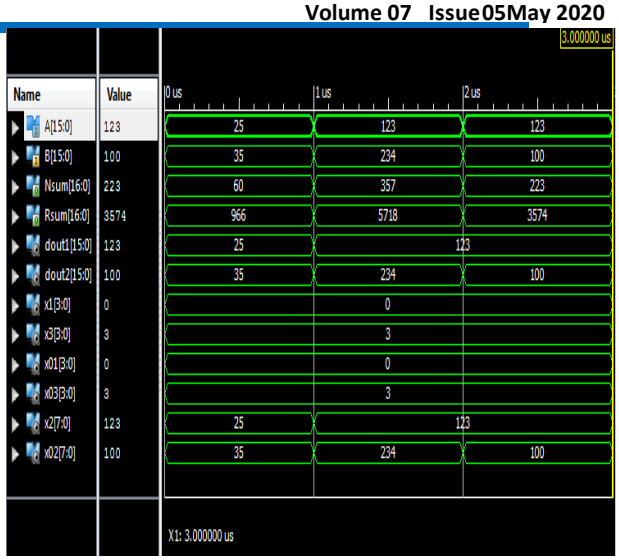

Fig 13: simulated wave forms

Parameter	modulo adder with logic gates	modulo adder with reversible logic gates
No of LUTs	124	64
Power (mW)	1.012	0.522

Table 1: Parameter comparison table

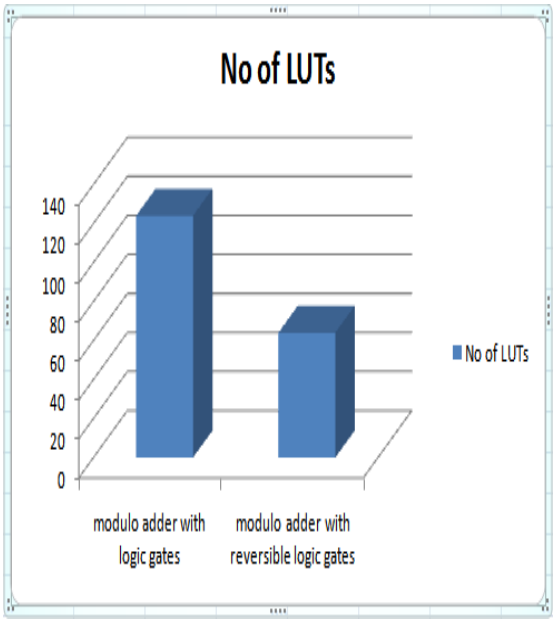

Fig14: LUT comparison bar graph

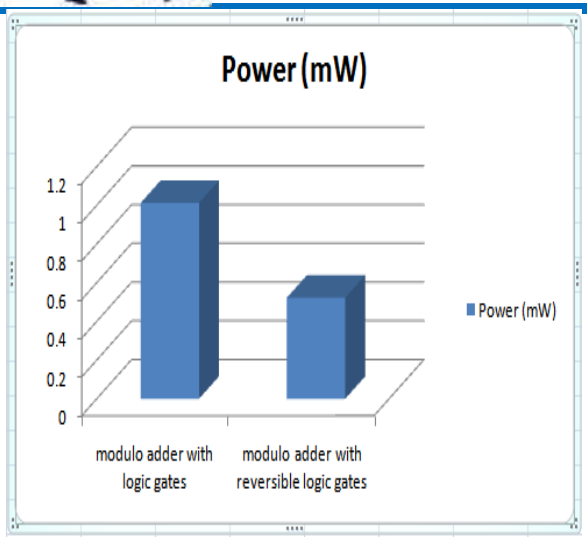

Fig15: power comparison bar graph

V. CONCLUSIONS

In this project, the proposed modular adder using reversible logic gates and bruntkung adder better than the existing modular adder using logic gates and koggestone adder. From the table 7.1 the proposed design uses less LUTs 64 when compared with the existing design with an LUTs count of 124.At the same time the power consumed for the proposed design is 0.522mW which is far better than the existing design power consumption with a value of 1.012mW, those are shown in table7.1.This work presents the reversible design of modular adders working on residue number system, a basic and RNS-based architectures. It is shown that a modulo 2*n*-1 parallel-prefix adder can be designed using small overheads over regular prefix adders.

REFERENCES

[1] T.M. Conte, E.P. DeBenedictis, P.A. Gargini, and E. Track, "Rebooting Computing: The Road Ahead," *Computer*, vol. 50, no. 1, pp. 20-29, 2017.

[2] M. Alioto (Ed.), *Enabling the Internet of Things: From Integrated Circuits to Integrated Systems*, Springer , 2017.

[3] A.S.Molahosseini, L.Sousa and C.H. Chang (Eds.), Embedded Systems Design with Special Arithmetic and Number Systems, Springer, 2017.

[4] C.H. Chang, A.S.Molahosseini, A.A.Emrani Zarandi, and T.F.Tay, "Residue Number Systems: A New Paradigm to Data path Optimization for Low-Power and High-Performance Digital Signal Processing Applications," *IEEE Circuits and Systems Magazine*, vol. 15, no. 4, pp. 26-44, 2015.

[5] L. Sousa, S. Antão, and P. Martins, "Combining Residue Arithmetic to Design Efficient Cryptographic Circuits and Systems," *IEEE Circuits and Systems Magazine*, vol. 16, no. 4, pp. 6-32, 2016.

[6] E.P. DeBenedictis, J.K. Mee, and M.P. Frank, "The Opportunities and Controversies of Reversible Computing," *Computer*, vol. 50, no. 6, pp. 76-80, 2017.

[7] R. Chaves and L. Sousa, "Improving RNS multiplication with more balanced moduli sets and enhanced modular arithmetic structures,"*IET Computers & Digital Techniques*, vol. 1, n. 5, pp. 472-480, 2007.

[8] A. Hiasat, "An Efficient Reverse Converter for the Three-Moduli Set (2*n*+1-1, 2*n*, 2*n*-1)," *IEEE Transactions on Circuits and Systems-II*, vol. 64, no. 8, 2017.

[9] A.S. Molahosseini, K. Navi, C. Dadkhah, O. Kavehei, S. Timarchi, "Efficient reverse converter designs for the new 4-moduli sets {2*n*–1, 2*n*, 2*n*+1, 22*n*+1–1} and {2*n*–1, 2*n*+1, 22*n*, 22*n*+1} based on new CRTs," *IEEE Transactions on Circuits and Systems-I*, vol. 57, no. 4, pp. 823-835, 2010.

[10] P. Patronik and S.J. Piestrak, "Design of Reverse Converters for General RNS Moduli Sets {2*k*, 2*n*-1, 2*n*+1, 2*n*+1-1} and {2*k*, 2*n*-1, 2*n*+1, 2*n*-1-1} (*n* even)," *IEEE Transactions on Circuits and Systems-I*, vol. 61, no. 6, pp. 1687-1700, 2014.