SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V13I6P106

Volume 13 Issue 6, 78-90, June 2026

Original Article

Design and Analysis of Reversible Logic based Forward and Reverse RNS Converters

Sunita Shirahatti[1], Rajashekhar Shettar[2]

[1]*ECE Department, JSS Academy of Technical Education, Affiliated to Visveswaraya Technological University, Karnataka, India.*
[2] *School of Electronics and Communication, KLE Technological University, Karnataka, India.*

[1]*Corresponding Author : sunithalshirahatti@jssateb.ac.in*

Received: 14 March 2026 Revised: 13 April 2026 Accepted: 12 May 2026 Published: 29 June 2026

Abstract - *In this paper, Residue Number Systems (RNS) forward and reverse conversion structures for moduli set {2^n+1, 2^n, 2^n-1} using reversible logic are presented. The conversion structures are based on arithmetic adders implemented using reversible logic without using any Lookup Tables. Among many power optimization methods, Reversible Logic (RL) is one of the emerging techniques used to build low-power arithmetic units. In this paper, the Reversible HNG gate and TKN gate are employed to implement different arithmetic adder structures, such as ripple carry adder, carry save adder, carry skip adder, parallel prefix adder, and multiplier. These arithmetic structures are used in the implementation of Forward and reverse converters. The Forward and Reverse converters are implemented and simulated in Cadence, and performance is analyzed for parameters area, delay, and power. The performance of reversible converters is compared with the conventional Forward and reverse converters. Experimental results shows that reversible logic implementation shows promising results in terms of power efficiency compared to conventional forward converter, as the dynamic range increases, and the carry save adder is the best suitable adder structure for the forward converter. The reversible CSA is approximately 35% smaller, 19% fewer gates, 32% lower power, and 22% faster than the conventional CSA. Reversible logic exploits the parallel nature of a CSA perfectly. The Reverse converters are also implemented using CRT and MRC methods using reversible gates and compared with the conventional implementations with respect to the area, delay, and power. The reversible gate implementation of CRT is approximately 2.3% slower than its conventional counterpart and 1.15% faster than the MRC method. CRT utilizes approximately 6.9 % less power with reversible gates compared to conventional gate implementation, and 1.1% less power compared to MRC, approximately.*

Keywords - *Chinese Remainder Theorem, HNG gate, Mixed Radix Conversion Reversible Logic (RL), Residue Number System, TKN gate.*

1. Introduction

The growing need for computers that are fast and use less energy has made people more interested in new design methods and different number systems. The Residue Number System (RNS) is well-known for being able to do arithmetic computations without forwarding the carry and in a parallel way. This makes calculations much faster and uses less power. In RNS, a whole number is represented by a group of residues that are pairwise relatively prime to a group of moduli. This structure makes it possible to do addition, subtraction, and multiplication operations separately in each residue channel, which makes parallel processing more efficient. RNS finds its application in filtering, convolution, FFT calculation, and cryptographic systems.

These real-world applications demand quick conversion from the binary number system to residue number representation and vice versa. These converters are called forward and reverse converters. The effectiveness of these converters has a big effect on how well an RNS processor works. Choosing the right moduli set is very critical for figuring out how sophisticated the hardware and how wide the dynamic range of the system will be. The three-moduli set {2^n-1, 2^n, 2^n+1} is a balanced set and is good for fast arithmetic operations.

The Residue Number System (RNS) is employed in digital signal processing applications to enhance the efficacy of computation and reduce power consumption. This numerical system is unconventional in that it employs remainders to represent numbers. It facilitates concurrent and rapid processing through its carry-free methodology [1]. The set of moduli that supports each specific RNS is the basis for the discrete execution of the fundamental arithmetic operations across multiple channels. The RNS is a non-weighted number system that employs a set of pairwise coprime integers known as moduli to represent integers. This is particularly advantageous in the fields of cryptography and digital signal processing because of its capabilities to simplify arithmetic operations and promote parallelism.

Let {m1, m2..., mn} be a set of integers that indicates a moduli set where all moduli are pairwise coprime. An integer X can be written as {x1, x2..., xn}, a set of residues where each xi=X mod mi. and the dynamic range of the RNS is given by D = m1 x m2 x….m_n. Any integer X value in the range 0 ≤ X <D can be uniquely represented.

In the Residue Number System, arithmetic operations are performed independently on each residue, which allows for high-speed parallel processing [2]. Given two numbers A and B with their residue representations:

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

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

Arithmetic operations in RNS are performed as follows:

$$C = (c_1 , c_2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots a_n)$$

Where,

$$Ci = (a_i + b_i) \; mod \; m_i$$

$$D = (d_1 , d_2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots d_n)$$

Where,

$$Di = (a_i - b_i) mod \; m_i$$

$$E = (e_1 , e_2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots e_n)$$

Where,

$$Ei = (a_i - b_i) mod \; m_i$$

To construct an RNS-based processing system, it is necessary to convert an input binary number into an equivalent residue number and vice versa. Consequently, RNS is commonly employed in computationally intensive and power-hungry applications like filtering, convolution, correlation, FFT computation in DSP, and cryptography [3].

Continued work on RNS started with three moduli sets, and to increase the dynamic range, alternate moduli sets have been proposed. Use of larger moduli sets allows smaller operands to be used for building compact, faster, and lower-power arithmetic units. Different moduli sets are discussed and highlighted in different articles [4]. Generally, {$2^n - 1$, 2^n, $2^n + 1$} is used for the implementation of converters and RNS arithmetic units for hardware simplification. To handle RNS data, it is essential to first convert the data in binary into RNS. The binary to RNS conversion is a forward converter, and from RNS to binary representation is a reverse converter [5]. It is necessary to implement efficient algorithms for the forward conversion to make them effective. The typical RNS processor is shown in Figure 1.

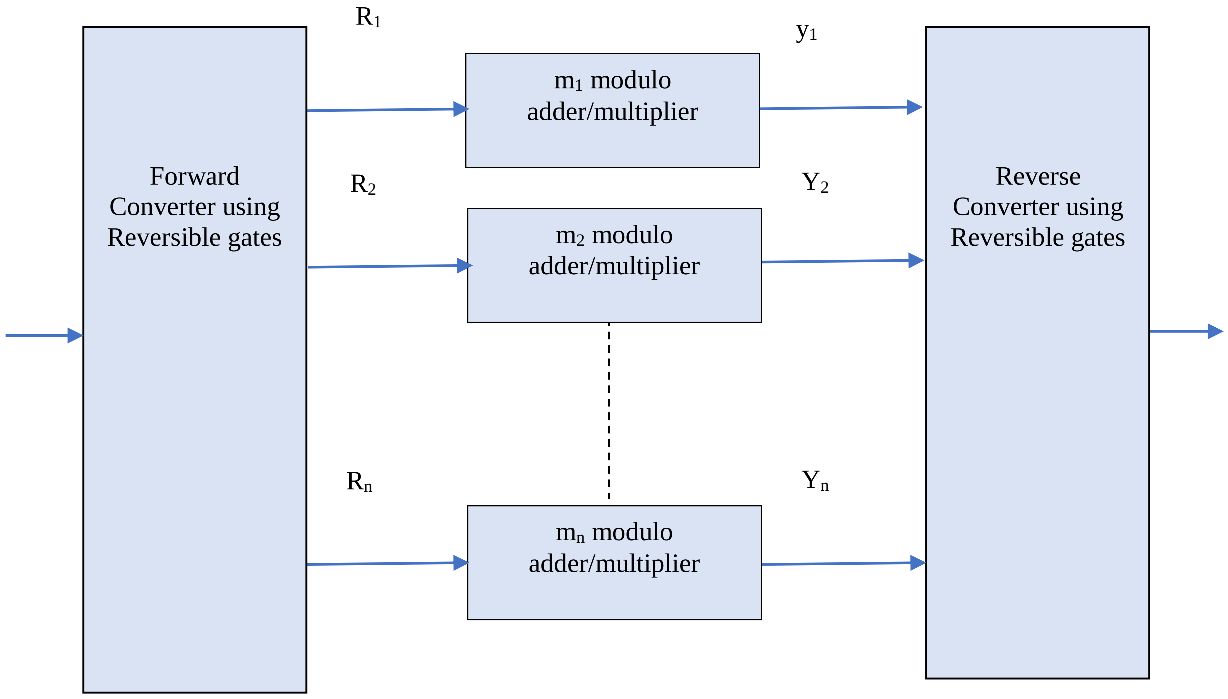

Fig. 1 Typical RNS processor

Modern applications require high-performance integrated circuits that can dissipate significant heat. To meet these demands, reversible logic is being used as an alternate design paradigm for low-power digital devices, complementing RNS research. It is based on Landauer's principle, which states that energy is lost when information is lost during irreversible computing. One way to avoid this problem is to make sure that there is a one-to-one correspondence between inputs and outputs in reversible logic. In principle, this keeps information from being lost and minimizes the amount of power that is lost. Reversible logic circuits have attracted considerable attention in areas such as low-power VLSI design, quantum computing, nanotechnology, and cryptographic hardware. Reversible logic implementation could potentially replace traditional design methodologies [6]. This type of logic, also known as energy recovery logic, avoids erasing a bit's information during its operation. Using reversible logic can improve device portability, reduce size, and simplify design. The construction of high-speed arithmetic circuits that are power-efficient has been the subject of numerous research projects that have been conducted independently using the Residue Number System and Reversible Logic. Reversible logic gates are employed to implement adder-based binary to RNS conversion.

The effective implementation of RNS converters still faces a number of obstacles, despite notable developments in reversible circuit design and RNS designs. For greater dynamic ranges, many binary-to-RNS converters now in use rely on Lookup-Table (LUT) based designs or traditional irreversible arithmetic structures, which raise hardware complexity and power consumption. Similar to this, big modulo adders or multipliers are frequently needed for reverse conversion methods like the Chinese Remainder Theorem (CRT) and Mixed Radix Conversion (MRC), which increases the size and time of VLSI implementations.

While reversible logic has been investigated for low-power arithmetic circuit design, little research has been done on the systematic design and comparative analysis of reversible logic-based forward and reverse converters using various adder architectures for the widely used moduli set $\{2^1-1$, 2üge, $2\omega+1\}$. Moreover, little research has been done on how various adder topologies affect the hardware performance of reversible RNS converters.

The objectives addressed in this paper are as follows:

- An adder-based binary to RNS converter using reversible logic gates is proposed.
- The performance analysis of the forward converter is carried out to identify the best adder architecture for different dynamic ranges.
- The reverse converter implementation is carried out using reversible logic gates, and performance parameters are analyzed.

The other sections of the paper are as follows. A review of related work on RNS converter architectures and reversible logic implementations is covered in Section 2. Section 3 describes the implementation methodology of the proposed reversible logic-based RNS forward and reverse converter architectures. Results and discussion, along with simulation setup, and performance evaluation of converter designs are shown in Section 4. Finally, the conclusion, limitations, and future scope in research is provided in Section 5.

2. Literature Survey

Residue Number System has drawn significant attention in Digital Signal Processing (DSP) and high-performance computing due to its ability to perform arithmetic operations in a carry-free and parallel manner. So, RNS is suitable for applications requiring high-speed computations and power consumption [7]. To use RNS, it is necessary to convert binary numbers to RNS and vice versa. This conversion is based on the set of moduli used. The moduli sets can be coprime or non-coprime. The process of computing residues of a binary number with respect to the moduli set is called a forward converter. Similarly, the process of combining residues to form a binary number is called a reverse converter. The moduli set selected implies the dynamic range and efficiency of the conversion process. Various moduli sets and architectural designs have been suggested in the previous work [8]. To simplify arithmetic operations and reduce hardware complexity, non-coprime moduli sets are used, which also provide a large dynamic range and efficient conversion process [9]. The moduli set is $\{2^n-1, 2^n, 2^n+1\}$ is frequently used, which offers balanced properties, is simple, and facilitates efficient arithmetic operation implementations [10]. For high-speed conversions and to reduce hardware requirements, a concept of pipelining is used [11]. Another approach involves the use of distributed data processing, which optimizes the conversion process by breaking it down into smaller, parallel operations [12]. To simplify the conversion process, the diminished-1 representation is used, which reduces the complexity of modulo operations. This method is particularly effective when used in conjunction with specific moduli sets, such as $\{2n-1, 2n, 2n+1\}$, and has been implemented successfully on FPGA platforms [13]. The performance evaluation of binary to RNS converters is carried out in terms of area, delay, and power consumption. Studies have shown that Carry-Save Adder (CSA) based designs are better in terms of speed, achieving approximately 20% faster operations [14]. To optimize power consumption in converters, researchers are employing reversible logic gates and implementing basic RNS operations, including addition and multiplication, with significant reductions in power usage [15]. In order to do backward conversion, all of them use either the Chinese Remainder Theorem (CRT) or Mixed-Radix Conversion (MRC). The MRC method works in a sequential way, while the CRT works in a concurrent way. One of the main problems with the CRT-based reverse converter is that it needs a big modulo adder at the end, and

MRC needs big multipliers. This issue is present in all converters that have been proposed in the literature. Ananda Mohan et al. [16] provide an explanation of the CRT and MRC, which are design principles and a theoretical context for RNS to binary conversion. The reverse conversion is one of the most significant and challenging operations.

The performance analysis of various hardware implementations of reverse converters, providing empirical insights into speed, area, and power metrics for different moduli sets and architectures [17]. In general, the VLSI implementation of reverse converters is complex and costly. So, optimized reverse conversion strategies leveraging core functions and residue data path optimizations are given by Patronik & Piestrak [18].

The application of full RNS variants in homomorphic encryption, necessitating efficient and secure residue-to-binary conversion schemes are explored in [19]. The reverse conversion component remains crucial for maintaining computational integrity in such sensitive applications. With rising interest in energy-efficient computing, reversible logic is being used for RNS conversion. In [20], the incorporation of reversible gates into RNS converter designs, significantly lowering power consumption and garbage output, is discussed. These works collectively pave the way for next-generation RNS-based systems in low-power, high-performance, and secure computing.

On the other hand, Reversible logic has emerged as a new field of research in computing that holds the promise of ultra-low power consumption by preserving information throughout the computation process. The concept of reversible logic is introduced by Landauer, R., which states that erasing one bit of information dissipates a minimum amount of energy, i.e., kT ln 2 in irreversible computations. Bennett, C. H., extended this idea and proved that computation can be made logically reversible to avoid dissipation of heat.

These foundational works laid the foundation for energy-efficient computation [21]. Various reversible gates, such as Feynman, Fredkin, Toffoli, Peres, and HNG gates, are used in the implementation of designs, including adders and sequential circuits, emphasizing their applications in low-power VLSI and quantum computing [22]. Many researchers were involved in designing new reversible gates for new functionality. Himanshu Thapliyal and M. B. Srinivas introduced the 4×4 TSG (Thapliyal-Srinivas Gate), a reversible gate capable of functioning as a full adder. Different circuits like the Arithmetic Logic Unit, adders, etc., were designed using TSG [23].

A substantial amount of research has been discussed in the literature about the synthesis of the reversible gates. Shende et al. proposed algorithms for reversible logic synthesis using the NOT, CNOT, and Toffoli gates [24]. Their contribution helped in realizing the implementations without ancilla inputs and reduced resource overhead. Saeedi and Markov [25] presented a survey of reversible circuit synthesis techniques supporting the work carried out so far. Two categories of synthesis techniques were presented.

The first category is based on exact approaches, which are optimal and computationally expensive, and the second one is based on heuristic approaches, which were scalable but not optimal [25]. Later, challenges involved in reversible logic synthesis are identified and presented by Naz SF et al. [26]. The basic challenge was to minimize the gate count and achieve the depth for large-scale circuits.

These challenges were addressed by a model-checking-based synthesis method, which was capable of synthesizing the circuits with minimal gate counts, but at higher computational costs. Then, a systematic truth table–based synthesis algorithm was proposed, reducing gate counts while simplifying circuit construction [27]. Reversible logic gates have been used in the design of control units and arithmetic circuits, functional units of processor architectures, resulting in significant reductions in power consumption [28].

Chiwande and Dakhole proposed an energy-efficient reversible multiplier optimized for low-power applications, demonstrating that reversible logic can effectively meet the demands of modern energy-constrained systems. Reversible gates can be used in practical applications ranging from low-power VLSI and quantum computing to arithmetic units, processor design, nanotechnology, cryptography, and fault-tolerant systems.

Their ability to minimize power dissipation and information loss positions them as a key technology for energy-efficient computing systems [30]. As the demand for energy-efficient architectures continues to rise, continued advancements in this field are expected to play a pivotal role in the design of next-generation low-power computing systems.

3. Materials and Methods

3.1. Forward Converter

The forward conversion stage is considered to be significant in the Residue Number System. The moduli employed in forward converters can be used to categorize them into two groups. The first category of Forward converters uses arbitrary moduli set and are typically constructed using look-up tables.

Special moduli-sets are employed by forward converters in the second category. The forward conversion architectures are simplified through the use of special moduli-sets, such as $\{2n-1, 2n, 2n1\}$. Figure 2 illustrates the block diagram for the forward converter that employs adders.

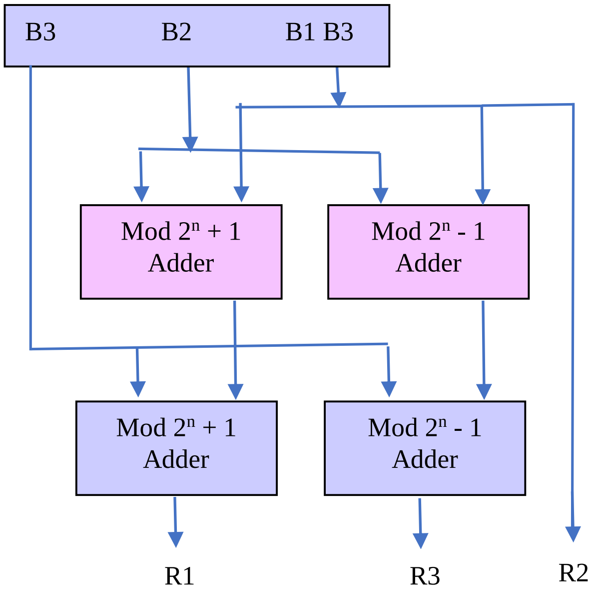

Fig. 2 Block diagram of forward converter

An algorithm used to convert a binary number into RNS using special moduli-set $\{2^n - 1, 2^n, 2^n + 1\}$ is presented below. Initially, divide the number into 3 blocks, B1, B2, and B3, with n bits each. Each n-bit number is represented as

$$B1 = \sum_{j=2n}^{3n-1} xj\ 2^{j-2n} \quad (1)$$

$$B2 = \sum_{j=n}^{2n-1} xj\ 2^{j-n} \quad (2)$$

$$B3 = \sum_{j=0}^{n-1} xj\ 2^{j} \quad (3)$$

The residue R2 shown in Equation (4) is simply n LSB bits, and can be obtained by right shifting by n-bits, and Residue R1 and R3 are given by the Equations (5) and (6) [31].

$$R2 = B3 \quad (4)$$

$$R1 = |B1 - B2 + B3|_{2n+1} \quad (5)$$

$$R3 = |B1 + B2 + B3|_{2n-1} \quad (6)$$

Different adder structures are employed in the conversion process. Research on improving the adder's delay performance and lowering power consumption is still ongoing. The basic adder topology used is Ripple Carry Adder (RCA), in which carry is propagated through all stages, and hence, the delay incurred is more. Carry Lookahead adder poses hardware complexity, increasing the area of implementation. The adder circuits can be implemented using reversible gates. Different reversible gates proposed in the literature are the Fredkin gate, Feyman gate, Toffoli gate, CNOT gate, Peares gate, TKS gate, and HNG gate. The basic structure, along with the logic function of a few reversible gates are shown in Figure 3.

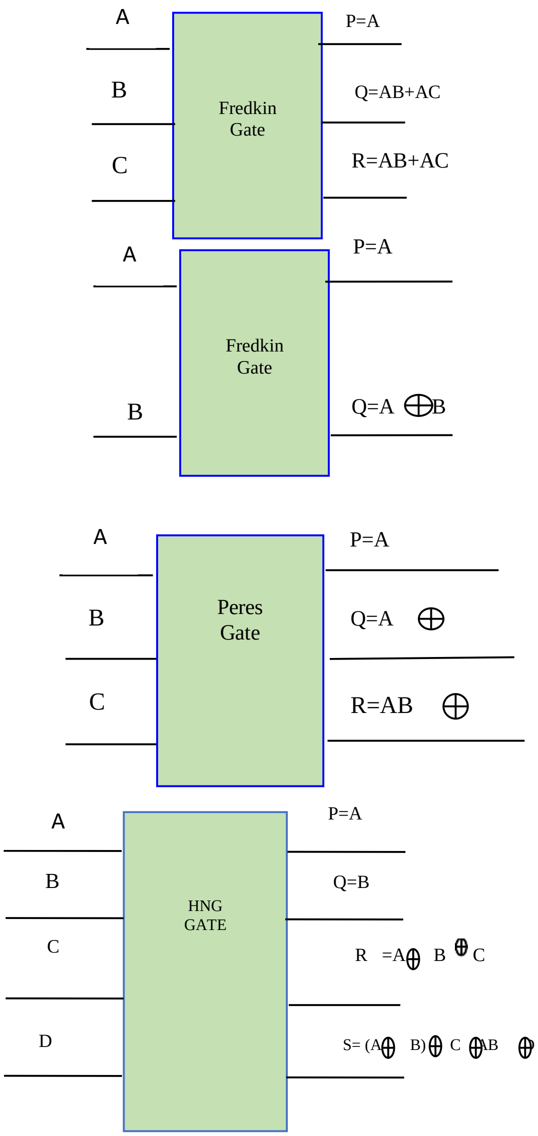

Fig. 3 Few reversible gates with logic function

The basic component of a ripple carry adder or any other adder is a full adder. The full adder functionality can be implemented using the Pearse gate or the HNG gate. Here, the HNG gate is preferred because quantum is 6.

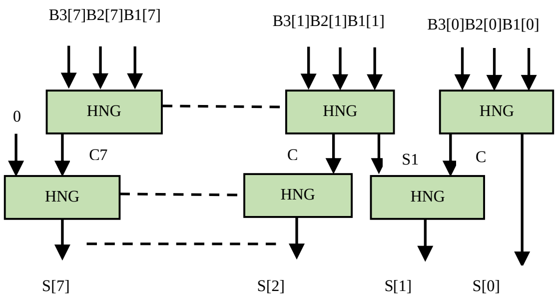

Fig. 4 Carry save adder block diagram using reversible HNG gates

If we use Peares gate for full adder implementation, then we need two gates whose overall quantum cost is 8. The full adder and 8-bit carry save adder implementation using the HNG gate is shown in Figure 4.

3.2. Reverse Converter

The RNS conversion to binary is an essential stage in RNS-based computing systems, particularly when interacting with conventional binary systems. This reverse conversion can be performed using CRT or MRC methods. The CRT method can be implemented in parallel, whereas the MRC method is sequential. The primary disadvantage of the CRT method is the necessity of a large modulo adder in the final stage, while the MRC method necessitates a limited number of large multipliers. Reverse converter implementation on VLSI is generally complex, expensive, and a constraint on RNS-based applications. The CRT definition goes as follows: Given moduli set $\{m1, m2\ldots, mk\}$ which are relatively prime and the residues are $(x1, x2\ldots, xk)$, then the result in binary is obtained using Equations (7) and (8) [32].

$$X = \left| \sum_{i=1}^{n} xi.Mi \, |Mi^{-1}|mi \; \right|M \qquad (7)$$

Where, $\mathrm{Mi} = \frac{M}{\mathrm{mi}}$ and $\mathrm{M_i^{-1}}$ is the multiplicative inverse of $\mathrm{M_i}$ with respect to mi such that

$$|M_i^{-1} {}^{*} 1|_{\mathrm{mi}} = 1,$$

Such that,

$$X = |m_1|M_1^{-1}\}_{m1}\, x1 + m_2|M_2^{-1}\}_{m2} X_2 + m_3|M_3^{-1}\}_{m3}\, X_3 \qquad (8)$$

The steps involved in the CRT algorithm are

i. The Mi's and their respective multiplicative inverse $\mathrm{M_1^{-1}}$, $\mathrm{M_2^{-1}}$, ……$\mathrm{M_n^{-1}}$ must be computed.
ii. Multiply and accumulate operations are performed
iii. Modular reduction is performed.

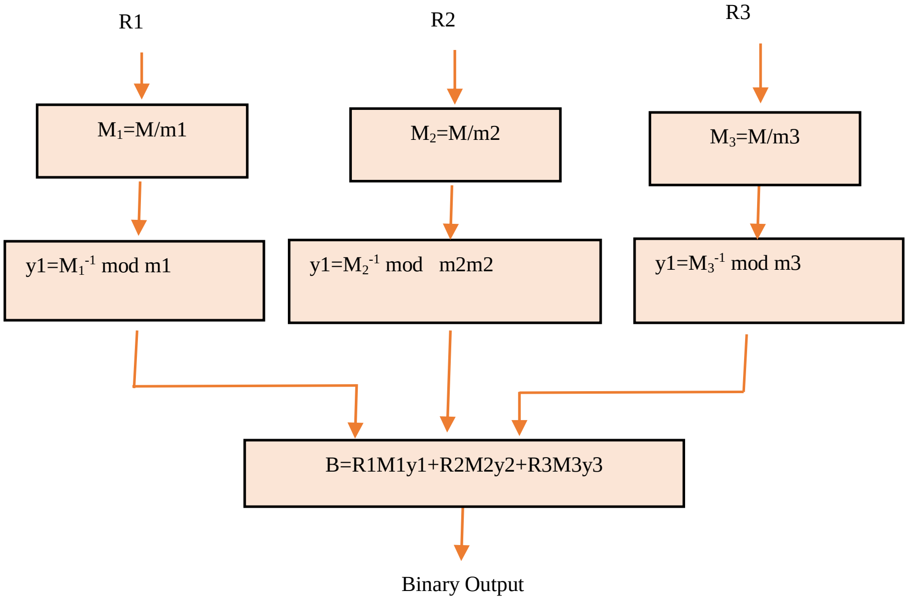

Fig. 5 Structure of the chinese remainder theorem conversion

The CRT operations are evidently parallel in nature, as shown in the schematic above. Therefore, the conversion process requires less time. The MRC conversion method serves as an alternative to the CRT method. The Mixed Radix Conversion structure is illustrated in Figure 6 and is further detailed in the mathematical equations below. This structure is based on a set of coprime moduli $\{m_1, m_2, m_3..., m_n\}$ and a residue (r).

4. Results and Discussion

The proposed architectures for the forward converter and reverse converter were implemented in Verilog HDL and validated by functional simulation. The designs were simulated and synthesized with the Cadence Ncsim simulator and Genus synthesis tool. The hardware implementations were assessed utilizing a 90 nm CMOS standard cell library. All synthesis tests were conducted at the usual process corner (TT) and a temperature of 25°C. Timing-driven synthesis was executed with a target clock frequency of 250 MHz. The efficacy of the proposed architectures was assessed utilizing typical hardware measures, including area, gate count, delay, power consumption, throughput, and energy per bit. The dynamic range parameter n was varied to evaluate the scalability of the forward converter architecture.

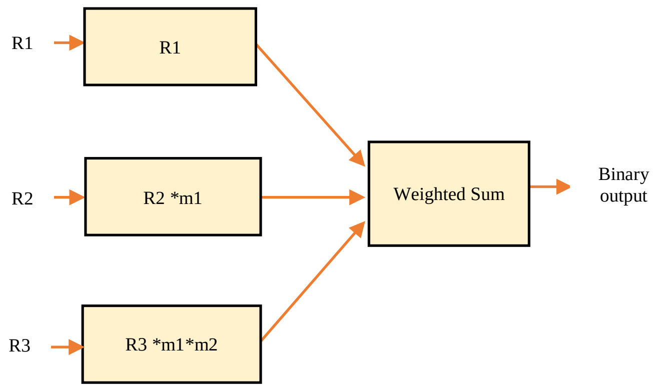

Fig. 6 Structure of mixed radix conversion

4.1. Performance Evaluation of Forward Converter

The Binary to RNS conversion is evaluated employing different adder topologies for the special moduli set $\{2^n-1, 2^n, 2^n+1\}$. In order to assess the impact of dynamic range on hardware performance, various values of n were taken into account. The synthesis results for forward converters implemented using conventional logic gates and reversible logic gates for different values of n are presented in Table 1.

From the table, the conventional implementation of the forward converter exhibits better area and speed for a dynamic ranges n=3 and n=4. But reversible implementation shows better results in terms of power consumption as the dynamic range increases ($n \geq 5$). This shows that the reversible logic has an impact on larger arithmetic circuits with respect to power consumption. The variation in parameters such as area, delay, and power with increasing dynamic range is shown in Figures 7, 8, and 9, respectively.

Table 1. Forward converter parameters for different values of n for moduli set $\{2^n-1, 2^n, 2^n+1\}$.

	Reversible gates				Conventional Gaes			
n	**Area**	**Gate count**	**Power**	**Delay**	**Area**	**Gate count**	**Power**	**delay**
n=3	610.061	81	1.35893e-04	2352	524.532	61	1.13283e-04	1953
n=4	877.247	116	2.10665e-04	2862	723.596	84	1.48105e-04	2680
n=5	1116.428	153	2.59541e-04	3288	1137.621	150	2.49219e-04	3336
n=6	1379.072	193	3.09693e-04	3613	1111.886	128	3.21128e-04	3300
n=7	1534.993	210	3.52587e-04	3704	1596.302	235	3.63519e-04	3762
n=8	1778.715	244	4.04669e-04	3826	1827.914	261	4.28939e-04	3760

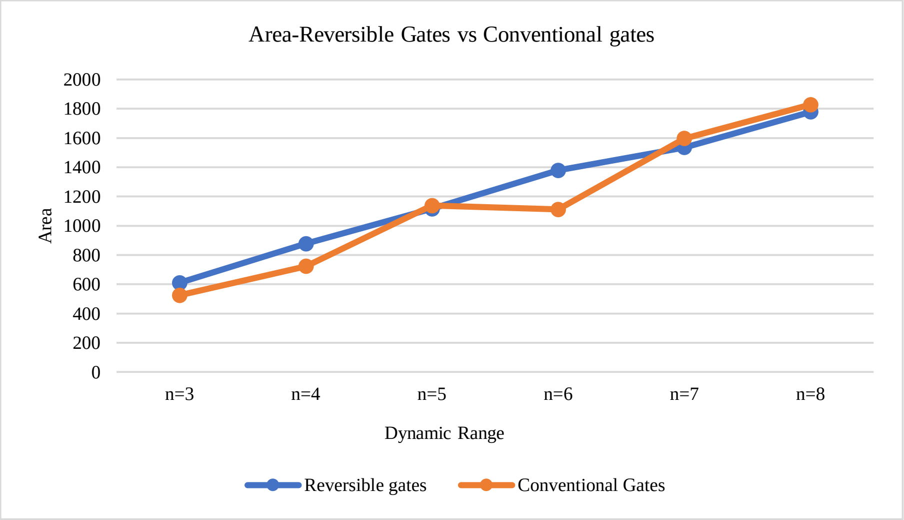

Fig. 7 Comparison of area parameter for different adder topologies

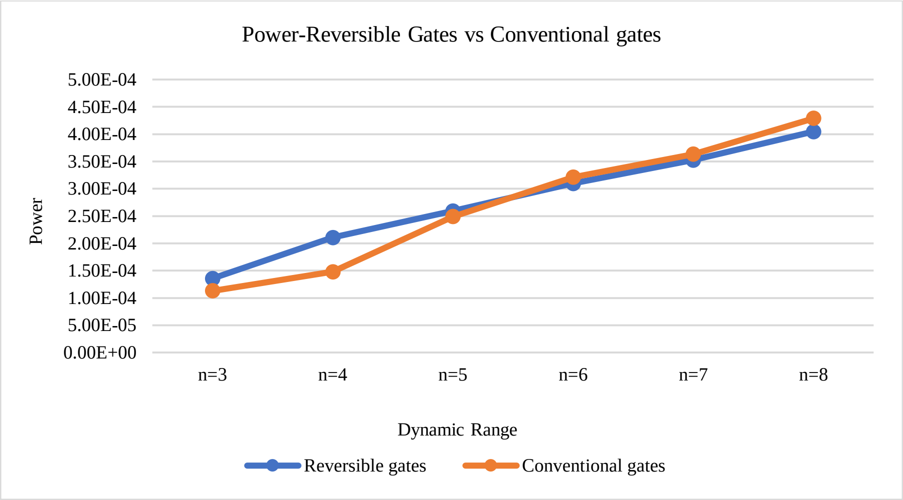

Fig. 8 Comparison of power parameters for different adder topologies

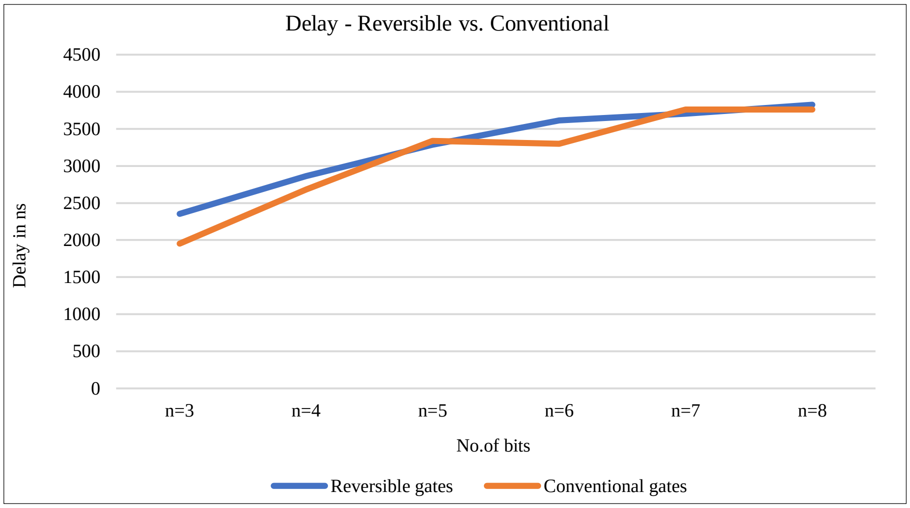

Fig. 9 Comparison of delay parameter for different adder topologies

4.2. Forward Converter with Different Adder Architectures

The performance analysis of the forward converter is carried out by employing different adder structures. Different adders such as RCA, CSA, carry skip, Carry Select, and Parallel Prefix Adder are used. All circuits were designed and synthesized keeping the dynamic range n = 8. The performance comparison of these adder architectures in terms of area, gate count, latency, and power consumption for both conventional gate implementation and reversible gate implementation is presented in Table 2. From Table 2, the Carry Save Adder (CSA) demonstrates better performance among the various adder structures used. Compared to the conventional CSA implementation, the reversible CSA implementation shows 35% reduction in area, a 19% reduction in gate count, a 32% reduction in power consumption, and a 22% improvement in delay. This improvement in results is the parallel carry handling capability of the CSA architecture. The comparative analysis of various adder architectures used in the forward converter is illustrated in Figures 10, 11, 12, and 13, respectively.

Table 2. Forward converter parameters using different adder topologies and n=8 for moduli set $\{2^n-1, 2^n, 2^n + 1\}$.

	Reversible gate implementation				**Conventional GAE implementation**			
	Area	Gate count	Power Watts	Delay (ps)	Area	Gate count	Power Watts	Delay (ps)
RCA	1778.715	244	4.04669e-04	3826	1827.914	261	4.28939e-04	3760
CSA	1179.250	135	2.25963e-04	2340	1818.831	167	3.33919e-04	2998
PPA	1765.091	252	4.14369e-04	3772	1597.816	193	3.32125e-04	3657
Carry kip	1788.55	237	3.98688e-04	3774	1827.157	254	2.40595e-04	2793
Carry select	1892.25	262	4.31857e-04	3856	1905.874	260	4.36297e-04	3858

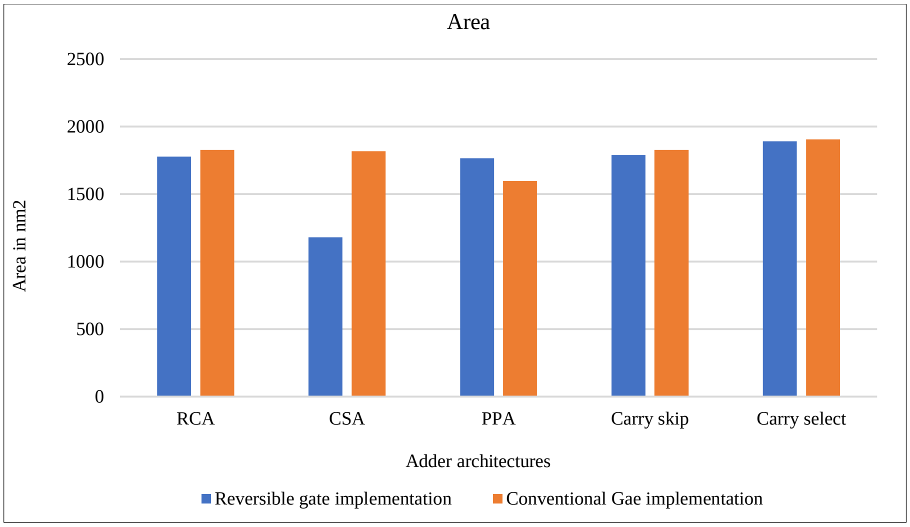

Fig. 10 Comparison of area parameter for different adder topologies

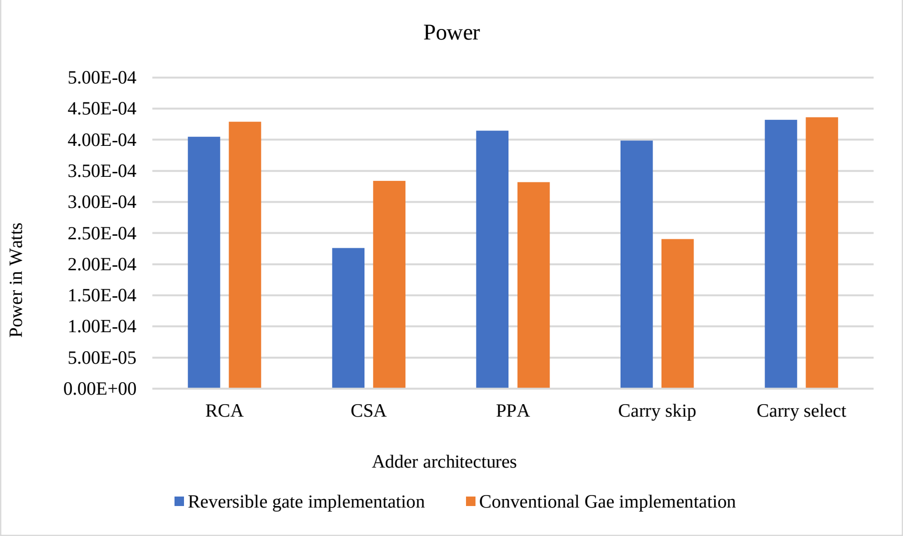

Fig. 11 Comparison of power parameters for different adder topologies

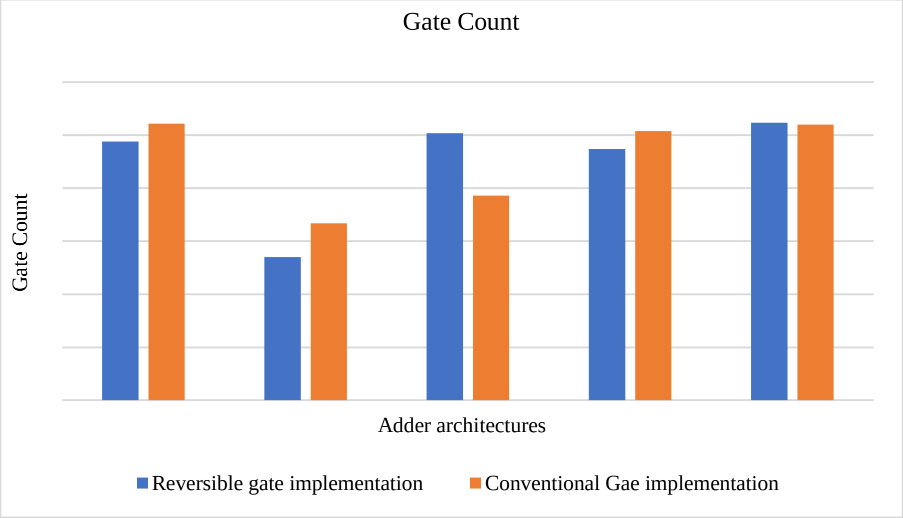

Fig. 12 Comparison of gate count parameter for different adder topologies

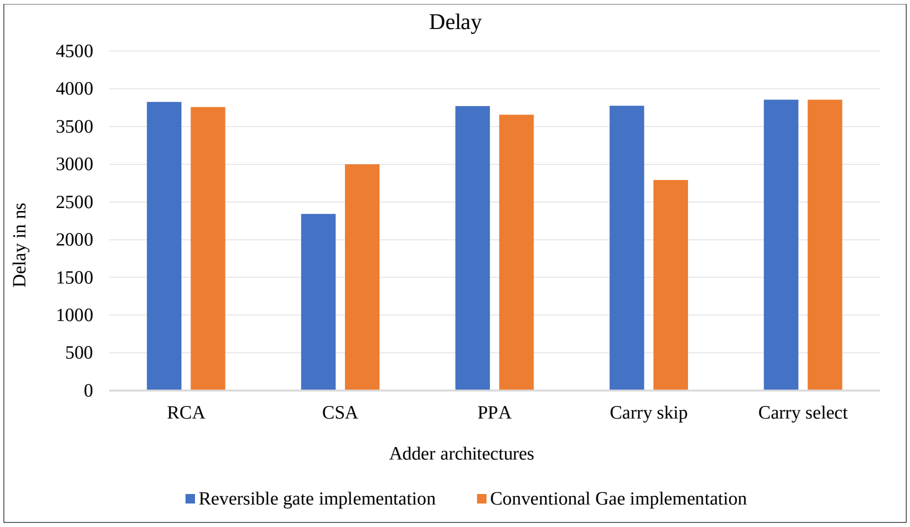

Fig. 13 Comparison of delay parameter for different adder topologies

4.3. Performance Evaluation of Reverse Converter

The conversion from RNS to binary is carried out using CRT and MRC methods. These implementations are carried out using both reversible and conventional gates. Table 3 lists area, power, and delay parameters for both methods. The synthesis results demonstrate that the CRT-based reverse converter using reversible gates exhibits 6.9% decrease in power compared to the conventional CRT implementation. Reversible CRT implementation performs a little better compared to the MRC technique in delay and power performance. But a slight area overhead is observed in both techniques. This area overhead is because of the constant input and outputs required in reversible circuits to preserve the input-output mapping. These are called ancilla inputs and garbage outputs. Table 4 shows converters designed with different moduli set in the past. The comparative analysis of the proposed forward and reverse converters against existing RNS-based converter designs across key performance parameters, namely area, delay, and power consumption, in the context of low-power circuit design.

Table 3. Comparison of performance parameters of RNS to binary converter for moduli set {2^n–1, 2^n, 2^n + 1}.

RNS to BIN conversion method	**Implementation using Conventional gates**			**Implementation using Reversible gates**		
	Area (µm2)	**Power (Watts)**	**Delay (ps)**	**Area (µm2)**	**Power (Watts)**	**Delay (ps)**
CRT	18680.291	3.19680e-02	82510	19550	2.97692e-02	84404
MRC	19050.605	3.83287e-02	95046	21310	3.29943e-02	98835

Table 4. Comparison of proposed converters with previous RNS-based converters

References	**Moduli Set**	**Type of Converter**	**Conventional / Reversible Implementation**	**Area**	**Delay**	**Power**
[20]	{2^{n-1}, 2^{n+k}, 2^{n+1}}	Forward + Reverse	Toffoli / Fredkin gates Quantum cost Reduction by 27%	------	------	--------
[34]	{2^{2n}, 2^{n-1}, 2^{n+1-1}} & {2^{n-1}, 2^{n+1}, 2^{2n+1}}	Hybrid converter	Fredkin gates + MRC Quantum cost reduction by 19.56 %	------	------	--------
[17]	{2^n–1, 2^n, 2^n + 1}	Forward + Reverse	Conventional gate implementation	Reduced area	Reduced Speed	------
[13]	{2^n–1, 2^n, 2^n + 1}	Forward Converter	Conventional gate implementation	Reduced area	Reduced Speed	------
[33]	{2^n–1, 2^n, 2^n + 1}	Modified CRT reverse converter	Conventional gates	area reductions up to 28.48%.	Delay reduction up tp 39.23%	Power Reduction up to 48.69%,

Proposed Forward Converter	$\{2^n-1, 2^n, 2^n + 1\}$	Forward Converter	Fredkin, Pares Feynamen, HNG gates	35% reduction in area,	22% increase in speed	32% less power consumption,
Proposed Reverse Converter	$\{2^n-1, 2^n, 2^n + 1\}$	Reverse Converter	Fredkin, Pares Feynamen, HNG gates	increased	2.3% slower	6.9 % less power

The work presented in [20, 34] focuses on the implementation of hybrid converters using reversible gates and concentrates on quantum cost reduction, but does not address area, delay, and power. The work presented in [13] and [17] uses moduli set $\{2^n-1, 2^n, 2^n+1\}$ and conventional gates and reports improvements in area and speed. Modified CRT-based reverse converter shown in [33] uses conventional logic gates and achieves area reductions of up to 28.48%, delay reductions of up to 39.23%, and power reductions of up to 48.69%. The proposed forward converter is implemented using reversible gates, and moduli set $\{2^n-1, 2^n, 2^n+1\}$, achieving a 35% reduction in area, a 22% improvement in speed, and a 32% reduction in power compared to conventional gate implementation.

The proposed reverse converter achieves a 6.9% reduction in power consumption with only a marginal 2.3% increase in delay and area overhead. The reduction in power is due to Landauer energy dissipation through zero information erasure, and the increase in speed is due to parallel operations and reduced switching nodes.

4.4. Limitations and Future Scope

The complexity of the reversible logic-based RNS architectures as the dynamic range increases. Reversible implementation may add an overhead on circuit depth, which can affect the overall efficiency despite reduced power. The basic limitation in the RNS system is the overhead introduced by the reverse converter, which in turn increases the area, delay, and power. In RNS, it is difficult to perform operations like comparison, scaling, and sign detection, which limits the use of RNS in a variety of applications. The reversible implementation of RNS converters has several limitations. An increase in dynamic range increases the hardware complexity for different moduli sets. The complexity of reverse conversion introduces additional hardware overhead and affects the performance. A few operations, such as sign detection, division, and comparison, are difficult to implement and computationally intensive. Additionally, moduli set selection involves trade-offs between dynamic range and implementation efficiency. Further overheads, such as garbage outputs and increased circuit depth, may arise, affecting overall design efficiency because of reversible gates. Future research may focus on implementing pipelined and parallel architectures. Furthermore, the implementation with new technology nodes such as 45nm, 28nm, and FinFET technology may help in improving the performance.

5. Conclusion

The design and implementation of reversible logic-based forward and reverse converters for the RNS System with moduli set $\{2^n-1, 2^n, 2^n + 1\}$ is presented. The RNS converters are built using various adders such as ripple Carry, carry save, carry skip, and parallel prefix, and a multiplier. Reversible gates such as HNG and TKN gates are used in the implementation of these adders. The performance analysis is carried out through simulations in Cadence 90nm technology. The reversible logic implementation of the forward converter demonstrated remarkable power efficiency as the dynamic range increases. The reversible CSA used in the forward converter exhibits 35% reduction in area, 19% decrease in gate count, 32% less power consumption, and 22% increase in speed compared to its irreversible counterparts. The reverse converters are implemented using CRT and MRC methods. The reversible gate implementation of the reverse converter using the CRT technique shows 1.1% faster and consumes less power compared to its MRC counterpart, and 2.3% slower than its irreversible counterpart. CRT technique consumes approximately 6.9 % less power with reversible gates compared to conventional gate implementation, and 1.1% less power compared to MRC, approximately. The efficiency of reversible gates in energy consumption applications makes them suitable for DSP, cryptography, and next-generation low-power VLSI systems.

References

[1] Amos R. Omondi, and Premkumar A. Benjamin, *Residue Number Systems: Theory and Implementation*, World Scientific, vol. 2, 2007. [CrossRef] [Google Scholar] [Publisher Link]

[2] Omar Abdelfattah, Andraws Swidan, and Zeljko Zilic, “Efficient Direct Analog-to-Residue Conversion Schemes,” *ICSES 2010 International Conference on Signals and Electronic Circuits*, Gliwice, Poland, pp. 85-88, 2010. [Google Scholar] [Publisher Link]

[3] W. Kenneth Jenkins, Michael A. Soderstrand, and C. Radhakrishnan, “Historical Patterns of Emerging Residue Number System Technologies During the Evolution of Computer Engineering and Digital Signal Processing, *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, Florence, Italy, pp. 1-5, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[4] Dina Younes, and Pavel Steffan, “A Comparative Study on Different Moduli Sets in Residue Number System,” *2012 International Conference on Computer Systems and Industrial Informatics*, Sharjah, United Arab Emirates, pp. 1-6, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[5] P.V. Ananda Mohan, *Forward and Reverse Converters for General Moduli Sets*, Residue Number Systems, Springer, Boston, MA, vol. 677, pp. 11-57, 2002. [CrossRef] [Google Scholar] [Publisher Link]

[6] M.C. Parameshwara, and M. Nagabushanam, "Novel Low Quantum Cost Reversible Logic based Full Adders for DSP Applications," *International Journal of Information Technology*, vol. 13, no. 5, pp. 1755-1761, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] V. Bavya, and R. Uthira Devi, "Optimizing the Precision of Digital Signal Processors using Residue Number System," *Imperial Journal of Interdisciplinary Research*, vol. 2, no. 4, pp. 1113-1112, 2016. [Google Scholar]

[8] P.V. Ananda Mohan, *Residue Number Systems*: *Theory and Applications*, 1st ed., Birkhäuser Cham, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[9] Mansour Bader, Swidan Andraws, and Al-Hadid Mazin, "New Non-Coprime Conjugate-Pair Binary to RNS Multi-Moduli for Residue Number System," *Computer Science and Information Technology*, vol. 105, pp. 105-111, 2017. [CrossRef] [Google Scholar]

[10] K. Vijaya Vardhan, and Sarada Musala, "A Fast RNS Processor for Moduli Set {2n+ 1, 2n, 2n-1}," *2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, Greater Noida, India, Greater Noida, India, pp. 1043-1047, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] Robert Smyk, and Maciej Czyżak "High-Speed Binary-to-Residue Converter Design using 2-Bit Segmentation of the Input Word," *Scientific Journal of Gdynia Maritime University*, vol. 124, pp. 42-56, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[12] Peter Boyvalenkov et al., "Classification of Moduli Sets for Residue Number System with Special Diagonal Functions," *IEEE Access*, vol. 8, pp. 156104-156116, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[13] Ivan Krstić, Negovan Stamenković, and Vidosav Stojanović, "Binary to RNS Encoder for the Moduli Set {2n− 1, 2n, 2n+ 1} with Embedded Diminished-1 Channel for DSP Application," *University Proceedings, Series: Electronics and Energetics*, vol. 29, no. 1, pp. 101-112, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[14] Shamim Akhter, Divya Bareja, and Satyendra Kumar, "Designing of Vedic based Modulo Multiplication in Residue Number System," *International Journal of Reconfigurable and Embedded Systems*, vol. 7, no. 2, pp. 67-73, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[15] Amir Sabbagh Molahosseini et al., "Towards Efficient Modular Adders based on Reversible Circuits," *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, Florence, Italy, pp. 1-5, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[16] P.V. Ananda Mohan, "Reverse Conversion using Core Function, CRT and Mixed Radix Conversion," *Circuits, Systems, and Signal Processing*, vol. 36, no. 7, pp. 2847-2874, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[17] Viktor Kuchukov et al., "Performance Analysis of Hardware Implementations of Reverse Conversion from the Residue Number System," *Applied Sciences*, vol. 12, no. 23, pp. 1-22, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] Piotr Patronik, and Stanisław J. Piestrak, "Design of RNS Reverse Converters with Constant Shifting to Residue Datapath Channels," *Journal of Signal Processing Systems*, vol. 90, no. 3, pp. 323-339, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[19] Jung Hee Cheon et al., "A full RNS Variant of Approximate Homomorphic Encryption," *Selected Areas in Cryptography – SAC 2018: 25th International Conference*, Calgary, AB, Canada, vol. 11349, pp. 347-368, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[20] Ailin Asadpour, Amir Sabbagh Molahosseini1, and Azadeh Alsadat Emrani Zarandi, "The use of Reversible Logic Gates in the Design of Residue Number Systems," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 2, pp. 2009-2022, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[21] Charles H. Bennett, "Notes on Landauer's Principle, Reversible Computation, and Maxwell's Demon," *Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics*, vol. 34, no. 3, pp. 501-510, 2003. [CrossRef] [Google Scholar] [Publisher Link]

[22] Ms. Surekha, and Mohd. Rashid Ansari, "Survey of Reversible Logic: A Review," *Grenze International Journal of Engineering and Technology (GIJET)*, vol. 8, no. 2, pp. 628-633, 2022. [Google Scholar]

[23] Himanshu Thapliyal, and M.B Srinivas, "A new Reversible TSG GATE and its Application for Designing Efficient Adder Circuits," *arXiv preprint*, pp. 1-5, 2005. [CrossRef] [Google Scholar] [Publisher Link]

[24] V.V. Shende et al., "Synthesis of Reversible Logic Circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22, no. 6, pp. 710-722, 2003. [CrossRef] [Google Scholar] [Publisher Link]

[25] Chua Shin Cheng, and Ashutosh Kumar Singh, "Heuristic Synthesis of Reversible Logic-A Comparative Study," *Advances in Electrical and Electronic Engineering*, vol. 12, no. 3, pp. 210-225, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[26] Syed Farah Naz, and Ambika Prasad Shah, "Reversible Gates: A Paradigm Shift in Computing," *IEEE Open Journal of Circuits and Systems*, vol. 4, pp. 241-257, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[27] Shaveta Thakral, Dipali Bansal, and S.K. Chakarvarti, "Review of Truth Table based Reversible Logic Synthesis Methods," *2015 International Conference on Soft Computing Techniques and Implementations (ICSCTI)*, Faridabad, India, pp. 164-169, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[28] Lafifa Jamal, Md. Masbaul Alam, and Hafiz Md. Hasan Babu, "An Efficient Approach to Design a Reversible Control Unit of a Processor," *Sustainable Computing: Informatics and Systems*, vol. 3, no. 4, pp. 286-294, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[29] Spoorti Patil et al., "Developing an Energy-Efficient Reversible Multiplier Architecture Utilizing TSG and Fredkin Gates," *ICT Analysis and Applications: Proceedings of ICT4SD*, Springer, Singapore, vol. 1162, pp. 243-254, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[30] Kanchan S. Tiwari et al., "Reversible Logic Gates and Applications-A Low Power Solution to VLSI Chips," *Mathematical Modelling of Engineering Problems*, vol. 11, no. 3, pp. 705-720, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[31] Omar Abdelfattah, "*Data Conversion in Residue Number System*," Thesis, McGill University, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[32] A.N. Babatunde et al., "Application of Residue Number System in Information Security: A Systematic Review," *Sule Lamido University Journal of Science and Technology*, pp. 195-219, 2025. [Google Scholar] [Publisher Link]

[33] Piotr Patronik, "On Reverse Converters for Arbitrary Multi-Moduli RNS," *Integration*, vol. 75, pp. 158-167, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[34] Ailin Asadpour, Amir Sabbagh, and Azadeh Emrani, "A Hybrid Forward/Reverse Converter in Reversible Logic to Reduce Hardware Complexity of Residual Number System," *Majlesi Journal of Electrical Engineering*, vol. 18, no. 2, pp. 1-15, 2024. [CrossRef] [Google Scholar] [Publisher Link]