ISSN- 2394-5125 VOL 08, ISSUE 03, 2021

HIGH PERFORMANCE RNS FOR QR DECOMPOSITION MATRIX INVERSION ARCHITECTURE

Mr.P.S.JAYAKUMAR 1, Mr.C.PALANI NEHRU 2, GANTA SREENU SUNDAR KUMAR REDDY

1Assistant Professor, Department of ECE,
2Assistant Professor, Department of CSE,
3 Student, Department of ECE,
1,2,3 Dhanalakshmi Sirinvasan College of Engineering and Technology, Chennai

Abstract: Field Programmable Gate Arrays (FPGA) is emerged in many Digital Signal Processing (DSP) applications with the inclusion of dedicated core processing elements as logical blocks. But the technology remains limited in its ability to support high speed demands which gives rise to unified arithmetic models for performing some core functional units. This paper presents high performance RNS (Residue Number System) arithmetic for Q-R decomposition (QRD) to perform matrix inverse core which limits the overall system performance of FPGA implementation of OMP algorithm for compressive sensing signal reconstruction. In this article is also introduced a new memory efficient on-chip Ram-based) reverse conversion unit capable of performing high speed RNS computation. This hardware-optimized RNS architecture can take wide range of input operand sizes with different sets of moduli sets. The design is implemented in an ALTERA FPGA Cyclone-II device. Experimental results proved that this memory efficient reverse conversion RNS architecture outperforms all other state–of–the–art FPGA implementations.

Keywords: Field Programmable Gate Arrays (FPGA), Residue Number System (RNS), modulo operation, Q-R decomposition (QRD).

1. Introduction

In recent years, residue number system (RNS) has been widely used for implementing many high-speed applications due to its properties like parallel processing, modular operation and carry-bound delay optimized arithmetic. However, inter-modular operation and conversion between residue to binary/ binary to residue number systems (and vice versa) are awkward in nature and this will limit its applicability in many applications ([1], [2]).

In RNS system some basic arithmetic operations, such as addition and multiplication, can be performed independently using dedicated residue channels which offer better optimization as compared to all other conventional 2's complement arithmetic models. In many cases, though the selection of the moduli sets is predominantly influences the overall system performance that decides the level of parallelism and attainable dynamic Ranges (DR), the design of reverse converters for all these moduli sets also consider as basic limitation factor.

The dynamic range is pre-determined from the product of all moduli from given moduli set. According to demands the dynamic range can be extended in two ways as follows: Vertical extension- Bit size of each moduli is increased. Horizontal extension-The number of moduli used is increased with improved level of parallelism. In both the cases the order of reverse converter complexity is increased linearly. In general, reverse conversion is computationally complex process and affects the system performance.

ISSN- 2394-5125 VOL 08, ISSUE 03, 2021

2. Related works

Though the Residue Number System (RNS) offers potential advantages with its ability to carry out fast carry-free arithmetic, still the computation complexity and memory requirement tend to investigate this system. In many existing works path delay is considered as a bottle neck to attain achievable system performance. Though several works investigate wide range of delay optimization, the problems associated with reverse conversion stage [3] also influence in greater level when RNS system is implemented in practice.

Most of the previous works investigate [4], [5] unified design for modulo multiplication and reverse conversion separately. However, even with efficient reverse conversion and residue computation, it is impossible to adopt RNS in practice for applications which involve complex processing elements like QR decomposition. Therefore, significant penalty gap exists in most of the previous RNS works [6] which consider unified system without considering features available in target devices. Hence, the statistical nature of end devices and associated constraints are the prime considerations to carry out design optimization and efficient full RNS systems, so that it can give prominent results in all kind of applications. Moreover, to achieve device restricted RNS systems, one should also consider the systematic requirement of end application where the RNS arithmetic is going to be implemented, need to be considered. In general, compared to moduli residue operations, conversion modules are considered as essential factors for improved hardware realizations and all other optimizations.

In [7] mitigated the dynamic reprogramming requirement of any memory based reverse converters over different moduli sets by introducing memory-less binary conversion. This memory-less model derived from periodicity properties of residues to compute modular exponentiation. In [8] developed a memory-less reverse conversion in order to accommodate wide range of moudli sets in RNS computation to support large dynamic ranges without causing any significant complexity issues. In [9] developed fully generic reverse conversion structure to compute the binary conversion effectively which can accommodate large moduli sets. And also developed a fully generic reverse conversion structure, accommodating large moduli sets, to compute the binary conversion effectively.

In [10] proposed unique functional modes based on inherent relationship between the binary number and Chinese remainder theorem. Here, diagonal function is used to perform residue number system (RNS) to binary conversion and monotonic function for reverse conversion process. In [11] developed an efficient residue-to-binary converter with sign detector which makes use of sign value of different moduli sets. In [12] developed encryption and decryption in digital communication with Shannon fano compression using Residue Number System (RNS). Also pipelining technique is used to narrow down the hardware utilization rate during decoding and carry out validation checks on all the residue results for error free computation. In [13] binary number to residue number converter is proposed for modular multiplication with least possible hardware complexity and energy consumption. This method also invented pre-loaded product elements to narrow down the hardware cost and path delay during partial product generation for each MAC operation. In [14] novel end-around carry units (EAC) is developed to optimize the computational complexity in

ISSN- 2394-5125 VOL 08, ISSUE 03, 2021

modular addition to mitigate the performance trade off that exists in any RNS FIR filter design with increased FIR filter length.

The most effective approach for RNS computation which gives better optimization constitutes efficient arithmetic units called Processing Elements (PE) which can dynamically perform all operations or memory units to store pre-computed results for reverse conversion. In both the cases, the hardware complexity overhead and memory space requirements are major concerns which will increase when the dynamic range increases.

In this paper, a high performance RNS is proposed which includes both processing elements as well as memory units with the following advantages.

1) Distributed Arithmetic (DA) based residue computations are used for low complexity.

2) It can adopt RNS with any number of moduli sets to mitigate the dynamic range problems associated with existing RNS system.

3) Improved hardware utilization rate with level of parallelism.

In addition to complexity reduction, carry bound DA-driven modular multiplication is exploited for high speed RNS computation process. This work also exploits on-board block RAMs for residue to binary conversion process which are readily available in any recent FPGA devices. Here, the proposed RNS system is designed by considering basic dynamic range requirement of QR decomposition process and FPGA as a target end device.

3. RNS System

Residue number system is computed based on defined moduli set, which comprise of group of pair-wise prime integers {p0, p1,..., pn−1} as moduli's. The range of values (that the) RNS system can accommodate is depends on arithmetic model perform and size of moduli sets which is called dynamic range. A RNS system with dynamic range D can formulate [0, D − 1] numbers. In case of a particular modular operation, the actual arithmetic is performed as a series of independent channels in parallel and the number of channels directly depends on number of moduli sets. Hence, both dynamic range as well as the level of parallelism both depend on moduli sets. The proposed RNS incorporates Distributed Arithmetic canonical signed digit for RNS modular multiplication.

3.1 Moduli conversion

Considering input pair-wise prime moduli {p1,p2,…, pn} and a computed residue {m1,m2,…, mn} during forward conversion process of input integer value Z, i.e mi= |Z|. pi, follows Eq. (1):

$$|Z|P=|\sum_{i=1}^{n} mi|Pi\text{-}1|piPi|P \qquad (1)$$

Where P is the product of the pi's, and Pi=P/pi. According to word length constraints the input value Zis chosen based on dynamic range and the size of moduli sets also changed based on modular arithmetic. But for RNS MAC, multiplication is considered as end factor to derive input operand size and types of moduli sets.

ISSN- 2394-5125 VOL 08, ISSUE 03, 2021

Input operand Zis formulated as:

$$Z \triangleq \{m1, m2, \dots, mn\} \triangleq \{m1, 0, \dots 0\} + \{0, m2, \dots, 0\} + \{0, 0, \dots, mn\}$$
$$\triangleq Z1 + Z2 + \cdots + Zn \quad (2)$$

Hence, the reverse conversion process requires finding Zi's. The basic operations required to obtain all Zi is accomplished with an appropriate pre-computation process which derives inverse values and end product values of all moduli sets during reverse conversion process.

In most cases the residues of Zi are zeros except for mi. This exploits the fact that Zi is a multiple of pj where j≠i. Therefore, Zi can be expressed as:

$$Zi \triangleq m1 * \{0, 0, \dots 1, \dots, 0, 0\} \triangleq mi * Zi \quad (3)$$

Where, Zi is found such that |Zi|pi =1. This equation can be rewritten based on the relation between the number pi and its inverse pi-1, is as follows:

$$(piZpi\text{-}1) \bmod pi = 1 \quad (4)$$

Defining Pi as P/Pi, where $P = \prod_{i=1}^{k} mi$.

Then:

$$||Pi\text{-}1|piPi|pi = 1 \quad (5)$$

Since all pi's are relatively prime, the inverses exist:

$$Zi = \overline{|}Pi\text{-}1|\overline{pi}Pi \quad (6)$$

$$Zi = miZi = mi|Pi\text{-}1|piPi \quad (7)$$

$$Z = \sum_{i=1}^{n} Zi = \sum_{i=1}^{n} mi|Pi\text{-}1|piPi \quad (8)$$

To formulate RNS system for multiplication according to the expected final operand size can be easily derived statistically from both input operand size and moduli sets to keep the end results within the dynamic range.

3.2 Residue computation model

In general performance metrics of overall MAC units are largely depends on multiplication operation which decides the overall design complexity as well as leads worst case of critical path. Here hardware efficiency is accomplished through Distributed arithmetic modelling where partial products are computed without using any multiplication units.

In DA arithmetic model complex expressions are simplified using shift register and memory units. Here the proposed DA based approaches follows the principles of direct graph method, pre computed sign conversion results are used to generate end results without any intermediate states.

ISSN- 2394-5125 VOL 08, ISSUE 03, 2021

3.2.1. CSD model

Canonical Signed digit recoding technique is adopted to reduce the number of shift registers and adders required in shift based accumulation approach. CSD representation has two main properties such as:

1. Minimum numbers of non-zero digits which results in reduced number of additions.
2. No two consecutive digits are non zero which facilitates multiple representation for a single binary number.

3.3 On chip RAM based binary conversion

In RNS system the complexity level of memory constraints of any reverse converter is directly related to the size of the moduli set. For each moduliset formulated for RNS system, irrespective to the type of RNS arithmetic performed, reverse conversion is performed that produce the end results which are well within the dynamic range. Here, all the pre-computed reverse conversion results are stored and used as a basic core processing elements in reverse converter. During FPGA synthesis, these memory core PEs are converted into on-chip block RAMs or LUTs depends on the targeted FPGA devices. The attainable system performance of RNS system with integrated memory optimized reverse converter comes with considerable hardware complexity reduction with least possible path delay as shown in Fig. 1.

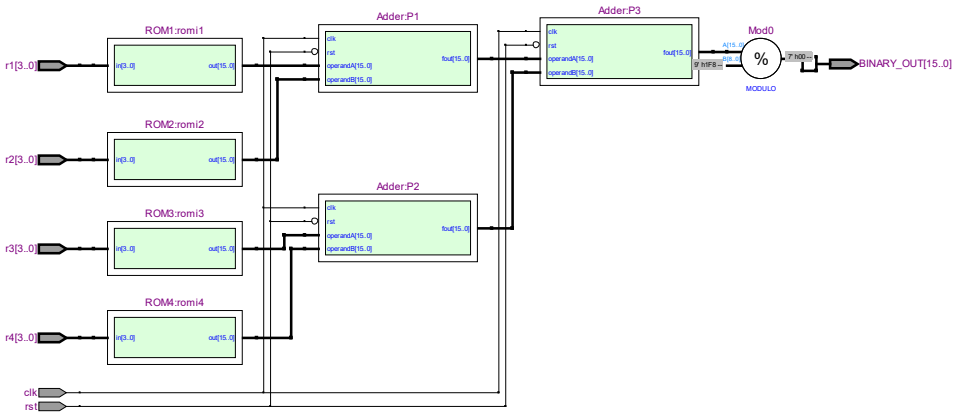

Figure .1Memory based RNS reverse converter design

4. Experimental Results

In order to facilitate efficiency of RNS system, experimental evaluation was carried outby considering user defined moduli sets {2n − 2, 2n − 1, 2n + 1, 2n - 1} with different sizes. All the design modules were scripted using Verilog HDL and its functionality is verified using exhaustive input stimulus (test bench). The proposed CSD-driven on-chip memory based RNS system is synthesized using QUARTUS II FPGA EDA tool with ALTERA CYCLONE II family devices as

ISSN- 2394-5125 VOL 08, ISSUE 03, 2021

shown in Fig.2. Here, the computational complexity is compared as configurable logical element. This paper focused on the implementation of high end FIR filters using optimized RNS units. The hardware synthesis results presented in this work proved that each level of optimizations carryout in RNS computation has a direct impact on hardware rate and performance retention of the FIR filter design. Here both the RAM based speculative reverse conversion and DA based residue computation used for path delay reduction in RNS system which can able to reduce the performance penalty gap in FIR filter design.

4.1 DA based residue computation

Here residue numbers multiplication through reduced number of adder and shifter register depth exhibits a better complexity reduction metrics than core RNS one. It is incorporated as follows: in DA computation all the non zero binary values and its weightages are computed identically and irrelevant to each other, end results is formulated through hierarchical shift/tree architecture.

The key merits of the proposed CSD methodology is the generation multiplied output using adder, such that it is consume lesser hardware than any other alternative. This system also eliminates the need of large sets of accumulators thereby reducing the hardware cost and increase the computational speed in all the possible ways.

The path delay retention in configurable moduli sets act as delay optimization RNS model as shown in Fig. 3. After the parameter evaluations, it is tested on various input operand size and its efficiency in path delay reduction is proved as shown in table .2.

ISSN- 2394-5125 VOL 08, ISSUE 03, 2021

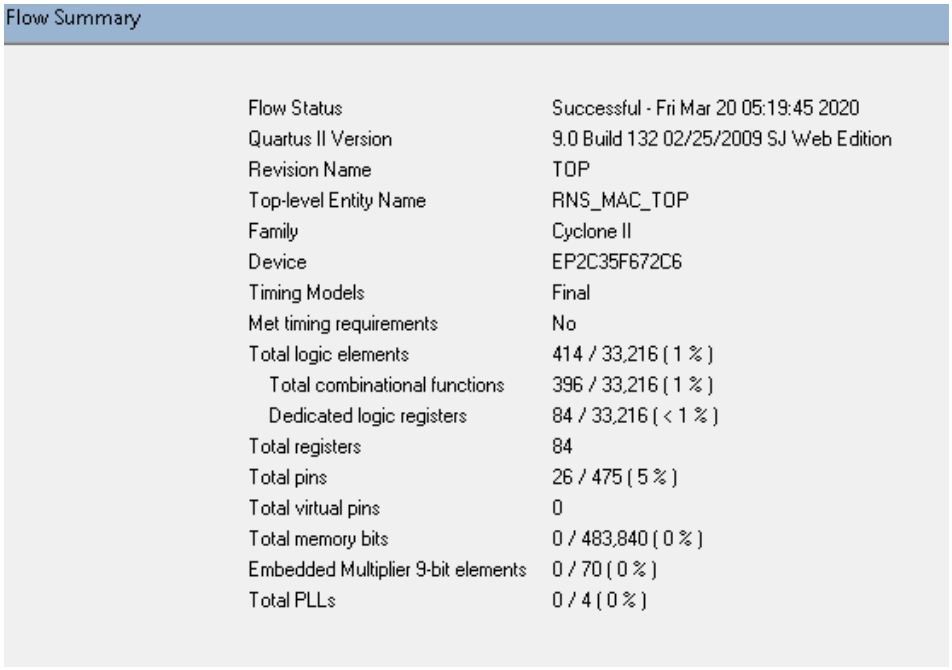

Figure .2 Resource utilization summary

Table 1. Performance metrics comparison of CSD RNS design

Residue computation model	RNS Model used	Input /Coeff. size	LEs used	Speed (MHz)
Multiplier based CSD RNS	Threemoduli set	8-bit unsigned	485	76.83
Proposed CSD RNS	Three moduli set	8-bit unsigned	395	76.78

4.3 Performance comparison of RNS with input operand size and Moduli sets

Here, optimizations were carried out based on target specific FPGA technology. The proposed reverse converter outperforms all other competitive arithmetic models in terms of path delay reduction. This is due to the fact that the proposed RNS requires fewer resources due to multiplier less CSD DA arithmetic as shown in Table 1. On the other hand, Field Programmable Gate Arrays (FPGAs) are
highly suitable for implementing memory-intensive reverse converters considered in this brief. Experimental results shown in Table 2 and 3 proved that the delay extension that arises due to extension of moduli set is negligible with some notable complexity overhead as compared to operand size extension.

Table 2. Complexity trade off comparison over input operand size

ISSN- 2394-5125 VOL 08, ISSUE 03, 2021

Input operand size	LEs used	Speed (MHz)
8-bit	485	76.83
16-bit	568	32.8

Table 3. Performance trade off comparison over moduli set extension

Moduli set level	LEs used	Speed (MHz)
Three moduli	485	76.83
Four moduli	414	75.18

Figure .3 Fmax report summary.

5. Conclusion

Finite hardware complexity tradeoff is a major bottleneck in the RNS design to meet real time dynamic range constraints. The methodologies presented in this work have mitigated that problem. In this work have been derived moderate level of operand size and dynamic range accumulation that involved in QR decomposition and designed RNS MAC accordingly. And this newly introduced RNS model provides reduced trade off hardware requirements with the moduli extension and

ISSN- 2394-5125 VOL 08, ISSUE 03, 2021

operand size variations. It also proved that the proposed CSD based DA model exhibits improved performance efficiency with considerable complexity reduction.

Reference

[1] Conway, Richard, and John Nelson. "Improved RNS FIR filter architectures." IEEE Transactions on Circuits and Systems II: Express Briefs 51, no. 1 (2004): 26-28.

[2] Pontarelli, Salvatore, Gian Carlo Cardarilli, Marco Re, and Adelio Salsano. "Optimized implementation of RNS FIR filters based on FPGAs." Journal of Signal Processing Systems 67, no. 3 (2012): 201-212.

[3] Xu, Minghe, Ruohe Yao, and Fei Luo. "Low-Complexity Sign Detection Algorithm for RNS {2 n-1, 2 n, 2 n+ 1}." IEICE transactions on electronics 95, no. 9 (2012): 1552-1556.

[4] Patel, Riyaz A., Mohammed Benaissa, and Said Boussakta. "Fast Modulo 2^{n}-(2^{n-2}+ 1) Addition: A New Class of Adder for RNS." IEEE Transactions on Computers 56, no. 4 (2007): 572-576.

[5] Navi, Keivan, Amir Sabbagh Molahosseini, and Mohammad Esmaeildoust. "How to teach residue number system to computer scientists and engineers." IEEE Transactions on Education 54, no. 1 (2010): 156-163.

[6] Molahosseini, Amir Sabbagh, FaeghehTeymouri, and Keivan Navi. "A new four-modulus RNS to binary converter." In Proceedings of 2010 IEEE International Symposium on Circuits and Systems, pp. 4161-4164. IEEE, 2010.

[7] Premkumar, A. Benjamin, E. L. Ang, and EM-K. Lai. "Improved memoryless RNS forward converter based on the periodicity of residues." IEEE Transactions on Circuits and Systems II: Express Briefs 53, no. 2 (2006): 133-137.

[8] Pettenghi, Hector, Ricardo Chaves, and Leonel Sousa. "RNS reverse converters for moduli sets with dynamic ranges up to $(8n+ 1) $-bit." IEEE Transactions on Circuits and Systems I: Regular Papers 60, no. 6 (2012): 1487-1500.

[9] Pettenghi, Hector, Ricardo Chaves, and Leonel Sousa. "Method to design general RNS reverse converters for extended moduli sets." IEEE Transactions on Circuits and Systems II: Express Briefs 60, no. 12 (2013): 877-881.

[10] Mohan, PV Ananda. "RNS to binary conversion using diagonal function and Pirlo and Impedovo monotonic function." Circuits, Systems, and Signal Processing 35, no. 3 (2016): 1063-1076.

[11] Hiasat, Ahmad. "A reverse converter and sign detectors for an extended RNS five-moduli set." IEEE Transactions on Circuits and Systems I: Regular Papers 64, no. 1 (2016): 111-121.

ISSN- 2394-5125 VOL 08, ISSUE 03, 2021

[12] Aremu, Idris Abiodun, and Kazeem AlagbeGbolagade. "RNS based on Shannon Fano coding for data encoding and decoding using {2n-1, 2n, 2n+ 1} Moduli Sets." Communications 6, no. 1 (2018): 25-29.

[13] Srinivasa Reddy, Kotha, and Subhendu Kumar Sahoo. "An approach for fixed coefficient RNS-based FIR filter." International Journal of Electronics 104.8 (2017): 1358-1376.

[14] Belghadr, Armin, and Ghassem Jaberipur. "FIR filter realization via deferred end-around carry modular addition." IEEE Transactions on Circuits and Systems I: Regular Papers 65.9 (2018): 2878-2888.