POZNAN UNIVERSITY OF TECHNOLOGY ACADEMIC JOURNALS
No 76 Electrical Engineering 2013

Assaid Othman SHAROUN*

RESIDUE NUMBER SYSTEM (RNS)

In the residue number system, a set of moduli which are independent of each other is given. An integer is represented by the residue of each modulus and the arithmetic operations are based on the residues individually. The arithmetic operations based on residue number system can be performed on various moduli independently to avoid the carry in addition, subtraction and multiplication, which is usually time consuming. However, the comparison and division are more complicated and the fraction number computation is immatured. Due to this, a residue number system is not yet popular in general-purpose computers, though it is extremely useful for digital-signal-processing applications. This thesis deals with the design, simulation and microcontroller implementation of some (residue number system based) building blocks for applications in the field of digital signal processing. The building blocks which have been studied are binary to residue converter, residue to binary converter, residue adder and residue multiplier.

1. INTRODUCTION

Residue number system is a non-weighted number system. So, it is very much different from the weighted number system like binary or decimal number systems. Residue arithmetic operations like addition, subtraction, and multiplication are inherently carry-free, i.e., each digit of the result is a function of only one digit from each operand, thereby independent of all other digits. This feature helps in considerable enhancement on the processing speed, which is the major criterion in DSP applications. This chapter deals with the fundamental concepts of the representation of numbers in residue system, the basic arithmetic operations, and the conversion of residue number into weighted number system using CRT and MRC process.

2. MATHEMATICAL EQUATIONS

2.1. Description of the residue representation

In a fixed-radix system, the number system is specified by its radix or base. Similarly, an RNS is completely described by stating the base, which consists of an n-tuple of integers rather than a single integer. Thus, the radix of an RNS is defined

* Zawia University, Libya.

by a set of integers m_1, m_2, …, m_n, where each individual member is called a modulus.

For any given base, the residue representation of an integer *x* is another *n*-tuple, (x_1, x_2, …, x_n), where x_i are integers defined by a set of *n* equations

$$x = q_i m_i + x_i \quad \text{for} \quad I = 1,2,\ldots,n \tag{1}$$

and q_i is an integer so chosen that $0 \le x_i \le m_i$. It is clear that q_i is the integer value of the quotient $\frac{x}{m_i}$. The quantity x_i is the least positive (integer) remainder of the division of *x* by m_i, and is designated as the residue of *x* modulo m_i or $|x|_{mi}$, often stated as *x mod mi.* The integer x_i is also called the *ith* residue digit of *x*. Here, *x* does not have to be a positive number but can be any integer. If *x* is negative, the quotient of $\frac{x}{m_i}$ will also be negative. By definition, $|x|_{mi}$ must be positive. The residue representation of a number is unique if the number lies within the dynamic range. For positive number *x*, the dynamic range lies between 0 to *M - 1,* where *M* is the product of all moduli. If both the positive and negative numbers are dealt with, the dynamic range extends from $-\frac{(M-1)}{2}$ to $\frac{(M-1)}{2}$ for *M* odd, and from $-\frac{M}{2}$ to $(\frac{M}{2}-1)$ for *M* even.

2.1.1. Additive inverse in Residue Aritmethic

For any integer *x* and modulo *m*

$$|-x|_m = |m-x|_m \tag{2}$$

Here $|m-x|_m$ is called the additive inverse of *x* modulo *m.* The additive inverse is very much useful in performing subtraction.

2.1.2. Multiplicative inverse in Residue Arithmetic

If $0 \le a \le m$ and $|ab|_m=1$, *a* is called the multiplicative inverse of *b* mod *m,* and is denoted by $a = \left|\frac{1}{b}\right|_m$. The quantity $\left|\frac{1}{b}\right|_m$ exists if and only if the greatest common divisor of *b* and *m* is equal to 1 and $|b|_m \neq 0$. The

multiplicative inverse is very important in RNS as it is used in Chinese Remainder Theorem and the Mixed Radix Conversion.

2.2. Residue Addition and Subtraction

For the residue system consisting of moduli m_1, m_2, …, m_n, let x and y be represented in residue form.

The residue addition or subtraction of x and y with respect to modulus M is given as:

$$\begin{aligned} x &\leftrightarrow (|x|_{m_1}, |x|_{m_2}, \ldots, |x|_{m_n}) \\ y &\leftrightarrow (|y|_{m_1}, |y|_{m_2}, \ldots, |y|_{m_n}) \\ |x \pm y|_M &\leftrightarrow (|\,|x|_{m_1} \pm |y|_{m_1}\,|_{m_1}, |\,x\,|_{m_2} \pm |\,y\,|_{m_2}|_{m_2}, \ldots, |\,|x|_{m_n} \pm |\,y\,|_{m_n}|_{m_n}) \end{aligned} \tag{3}$$

In this respect it is fundamentally different from any weighted number system, where a particular digit of the result is a function of the less significant digits of the operands. The absence of carry digits in residue addition provides the advantage of inherent high speed to the system.

2.3. Residue Multiplication

For the residue system consisting of moduli m_1, m_2, …, m_n, let x and y be represented by residue digits. Then the residue multiplication of x and y with respect to modulus M is given as:

$$\begin{aligned} x &\leftrightarrow (|x|_{m_1}, |x|_{m_2}, \ldots, |x|_{m_n}) \\ y &\leftrightarrow (|y|_{m_1}, |y|_{m_2}, \ldots, |y|_{m_n}) \\ |xy|_M &\leftrightarrow (|\,|x|_{m_1}|y|_{m_1}\,|_{m_1}, |\,x\,|_{m_2}|\,y\,|_{m_2}|_{m_2}, \ldots, |\,|x|_{m_n}\,|\,y\,|_{m_n}|_{m_n}) \end{aligned} \tag{4}$$

As Conversion to the Residue Representation

Equation (1) defines the residue of a number with respect to modulo m_i. By using this equation, it is always possible to compute the residue representation of a number. In a conventional computer this calculation would be carried out dividing x by m_i and determining the remainder. A far better method can be employed in a residue computer which is equipped to perform the residue operations of addition, subtraction, and multiplication with respect to modulo mi.

Let us consider an integer x expressed in binary form as

$$x = 2^n b_n + \ldots + 2^2 b_2 + 2b_1 + b_0$$

where the b_i's are the binary digits of x. Taking this expression modulo m_i we obtain

$$| x |_{m_i} = || 2^n |_{m_i} b_n + ... + | 2^2 |_{m_i} b_2 + | 2 |_{m_i} + b_0 |_{m_i}$$

If the powers of 2 modulo m_i are directly available (from the computer memory), $|x|_{mi}$ may be computed by merely adding (modulo m_i) those powers of 2 for which $b_i = 1$. Let the binary number 11001110 is to be converted to residue number with respect to modulo 3. It is clear that the residue of any odd power of 2 is 2, i.e

$$| 2^0 |_3 = 1, | 2^1 |_3 = 2 \quad \text{and} \quad | 2^2 |_3 = 1, \text{ etc.}$$

The number 11001110 has two coefficients of unity for even powers and three coefficients of unity for odd powers. Hence.

2.4. Conversion from Residue Number to Weighted Number System

Conversion from the RNS to weighted number system is a very important operation in RNS based system design. There are two basic approaches to the conversion, the Chinese Remainder Theorem and the Mixed Radix Conversion operation.

2.4.1. Chinese Remainder Theorem

Let the residue representation of x for moduli $\{m_1, m_2, \ldots, m_n\}$ is given as $(x_1, x_2, \ldots, x_n)$. The Chinese Remainder Theorem makes it possible to determine $|x|_M$, provided the greatest common divisor of any pair of moduli is 1. Using this theorem:

$$| x |_M = \left| \sum_{j=1}^{n} \hat{m}_j \left| \frac{x_j}{\hat{m}_j} \right|_{m_j} \right|_M \tag{5}$$

where: $\hat{m}_j = \dfrac{M}{m_j}$, $M = \prod_{j=1}^{n} m$.

2.4.2. Mixed Radix Conversion Process

In Chinese Remainder Theorem, arithmetic operations for modulo M are to be performed. The residue converters based on CRT are therefore, very much complicated. In contrast, the Mixed-Radix Conversion process requires arithmetic operations for modulo m_i only, thereby making all operations simpler as compared to CRT.

In MRC process, a number x is expressed in mixed-radix system. Suppose for moduli set $m_1, m_2, \ldots, m_n$, RNS representation of a number x is given as $(x_1, x_2, \ldots, x_n)$. The number x can be expressed in mixed-radix form as:

$$x = a_n \prod_{i=1}^{n-1} m_i + ... + a_3 m_1 m_2 + a_2 m_1 + a_1 \quad (6)$$

Where the a_i's are the mixed-radix coefficients. These a_i's are determined sequentially, starting with a_1, in the following manner:

Equation (6) is first taken in modulo m_1. Since all terms except the last are multiples of m_1, we have

$$| x |_{m_1} = a_1$$

Hence a_1 is just the first residue digit. To obtain a_2, first we subtract a_1 from x. The quantity $x-a_1$ is divided by m_1, and doing modulo operation with respect to m_2, we have

$$\left| \frac{x - a_1}{m_1} \right|_{m_2} = a_2$$

Similarly for a_3, $(a_2m_1+a_1)$ is subtracted from x. Dividing the quantity $(x-a_2m_1-a_1)$ by m_1m_2 and performing modulo operation with respect to m_3, we get

$$\left| \frac{x - a_2 m_1 - a_1}{m_1 m_2} \right|_{m_2} = a_3$$

In this way, by successive subtraction and division in residue notation, all the mixed-radix digits may be obtained.

2.5. Selection of Moduli

Selection of moduli and the number of moduli determine both the dynamic range and the complexity of the resulting hardware. There are certain points which must be considered for the selection of moduli. These points are:

- Moduli must be mutually prime.
- The magnitude of the largest modulus dictates the speed of arithmetic operations.
- All other moduli should be made comparable in size to the largest one.

The literature survey reveals that the most popular moduli set is: (2^n-1), 2^n, (2^n+1).

3. CONCLUSION

It is clear that RNS arithmetic provides parallel arithmetic operation for addition, subtraction and multiplication. Other operations like division, magnitude comparison, algebraic-sign determination and overflow detection are relatively difficult and complex with RNS. Therefore, RNS arithmetic finds extensive

applications in the field of signal processing where addition and multiplication are the predominant operations.

The main results of the dissertation are in the following:

1. investigations of possibility to use the moduli set for digital building blocks, discussions about the principial number *n*,
2. novel algorithm for conversion of negative binary number to residue form,
3. memoryless residue to binary converter circuit,
4. design and verification of the VHDL files for these building blocks.

REFERENCES

[1] N.Sszabo and R.J Tanaka, Residue Arithmetic and its Applications to Computer Technology, New York, McGraw-Hill, 1967.

[2] M.soderstrand et al., Residue Number System Arithmetic: Modem Applications in Digital Signal Processing, IEEE Press, NY, 1986.

[3] G.Alia, E.Martinelli,"VLSI binary-residue converters for pipelined processing" Computer J., Vol.33, no.5, pp.473-475, 1990.

[4] S.J.Piestrak, "Design of residue generators and multioperand modulo adders using carry-save Adders,"IEEE Trans. Comp., vol. 43, pp.68-77, Jan. 1994.

[5] A.B.premkumar,"Improved memory less RNS forward converter based on periodicity of residues, "IEEE Trans. Circuits and Systems-II, express Briefs, vol. 53, no.2, feb.2006, and pp.133-137.