ERROR CORRECTION OF CORRUPTED DATA USING A REDUNDANT RESIDUE NUMBER SYSTEM

Brian T. Hemmelman
Electrical and Computer Engineering Department
South Dakota School of Mines & Technology
Rapid City, SD 57701

Benjamin Premkumar
School of Computer Engineering
Nanyang Technological University
Singapore

Patel Anirudh Reddy
Electrical and Computer Engineering Department
South Dakota School of Mines & Technology
Rapid City, SD 57701

ABSTRACT

Data communicated by wired or wireless methods is subject to being corrupted from a variety of sources. When data is corrupted the errors not only need to be detected, but also they also need to be corrected so the data returns to its original and useful form. We have developed a technique that uses redundant residue number systems and the maximum likelihood principle to create an error detection and correction scheme that has the ability to detect up to R errors and correct up to R-1 errors if R redundant residues are used. The error correcting abilities of this approach have been demonstrated in hardware communication systems.

Keywords

Number Theory, Residue Number Systems, Information Theory, Communication Systems

INTRODUCTION

All electronic systems can be affected by various sources of noise and interference. This noise and interference can corrupt the signals and information that the electronic circuitry is manipulating. In digital electronic systems, the information is in the form of bits with the familiar values of '0' and '1'. Collections of bits are organized and used for various control, communication, and computational purposes. When the bits become corrupted by noise, the system may fail to correctly perform its assigned task. We have developed a tech-

nique based on the number theory of redundant residue number systems and the Maximum Likelihood Principle to allow for the detection and correction of corrupted digital data. For data encoded using R redundant residues, this method can detect up to R errors and correct up to R-1 errors. This error correction technique has been demonstrated in hardware communication systems.

METHODS

Unlike a fixed-radix system where a number is completely specified by stating the single fixed radix or base, in the residue number system the base does not consist of a single radix but instead consists of a number of integer moduli, m1, m2, m3, …, mN. The range of numbers that can be represented using this moduli set is then given by $\prod_{i=1}^{N} m_i$. Any given number to be represented then is defined by the residue set obtained after applying each of the moduli to the number. For example, consider a moduli set {m1, m2, m3}. Then a fixed radix number X would be represented by the residue set {r1, r2, r3} where r1 = X mod m1, r2 = X mod m2, and r3 = X mod m3.

For error detection and correction it is necessary that the moduli set is a set of relatively prime numbers. Here, relatively prime means that each modulus does not have to be a prime number itself but that amongst the moduli in the set the only common factor is one. The entire moduli set will now contain nonredundant and redundant moduli for the purposes of error detection and correction. The actual number can be represented by just the residues obtained using the nonredundant moduli. The product of just the nonredundant moduli now defines what is called the legitimate range, and the product of the entire moduli set now defines the illegitimate range.

As an example, the moduli set {5, 7, 8, 9, 11, 13} can be used where the subset {5, 7, 8} contains the nonredundant moduli and the subset {9, 11, 13} contains the redundant moduli. The legitimate range is thus 280, and the illegitimate range is 360360. For instance, the decimal number 220 then would be represented as the residue set {0, 3, 4, 4, 0, 12}. The subset {0, 3, 4} contains the nonredundant residues, and the subset {4, 0, 12} contains the redundant residues. This residue representation is what can then be used in a control or computational system or transmitted in a communication system.

When it is time to check if a computation was performed correctly or transmitted data was received properly, a reverse conversion of the residue set is made using the Chinese Remainder Theorem (Szabo, et. al. 1967). To reconstruct the original number from the residue set, first compute

$$M_i = \frac{M}{m_i} = \frac{\prod_{i=1}^{N} m_i}{m_i}. \quad \text{(Equation 1)}$$

So, using just the nonredundant moduli {5, 7, 8} from the above example yields

$M_1 = m_2 * m_3 = 7 * 8 = 56$,
$M_2 = m_1 * m_3 = 5 * 8 = 40$, and
$M_3 = m_1 * m_2 = 5 * 7 = 35$.

Next K_i is computed where K_i is the smallest positive integer multiple of M_i such that $(K_i * M_i) \bmod m_i = 1$. For the nonredundant moduli {5, 7, 8}, $K_1 = 1$, $K_2 = 3$, $K_3 = 3$. Note that the values of M_i and K_i are independent of the specific residues for any given encoded number. They depend only on the moduli set and thus are fixed values once the moduli set is chosen. Finally, the original data can be computed as

$$X = \left(\sum_{i=1}^{N} M_i * \left[(K_i * r_i) \bmod m_i \right] \right) \bmod M, \qquad \text{(Equation 2)}$$

where r_i are the residues for the specific data being converted.

For the decimal number 220 which has nonredundant residues {0, 3, 4} corresponding to the nonredundant moduli {5, 7, 8} the conversion is calculated as

$$X = \{(56 * [(1 * 0) \bmod 5]) + (40 * [(3 * 3) \bmod 7]) + (35 * [(3 * 4) \bmod 8])\} \bmod 280$$

$$X = [(56 * 0) + (40 * 2) + (35 * 4)] \bmod 280 = (80 + 140) \bmod 280 = 220.$$

A mixture of nonredundant and redundant residues will also produce the original data. As an example, if the residue subset of {0, 3, 4, 4} corresponding to the moduli subset of {5, 7, 8, 9} for the decimal number 220 is used, the new values of M_i and K_i are found to be $M_1 = 504$, $M_2 = 260$, $M_3 = 315$, $M_4 = 280$, $K_1 = 4$, $K_2 = 5$, $K_3 = 3$, $K_4 = 1$. The Chinese Remainder Theorem then yields

$$X = \left\{ \begin{matrix} (504 * [(4 * 0) \bmod 5]) + (360 * [(5 * 3) \bmod 7]) + \\ (315 * [(3 * 4) \bmod 8]) + (280 * [(1 * 4) \bmod 9]) \end{matrix} \right\} \bmod 2520$$

$$X = 2740 \bmod 2520 = 220$$

If one of the residues is corrupted by noise or an electrical fault, the redundant residue number system proposed has the ability to detect the error and correct it. In the previous example, perhaps the residue subset of {0, 3, 4, 4} has been corrupted into the values {3, 3, 4, 4} (i.e. r1 has been changed from the correct value of 0 to the erroneous value of 3). The conversion process then yields

$$X = \left\{ \begin{matrix} (504 * [(4 * 3) \bmod 5]) + (360 * [(5 * 3) \bmod 7]) + \\ (315 * [(3 * 4) \bmod 8]) + (280 * [(1 * 4) \bmod 9]) \end{matrix} \right\} \bmod 2520$$

$$X = 3748 \bmod 2520 = 1228.$$

However, the value 1228 is outside the legitimate range of 280 so the presence of the error has been detected.

The Maximum Likelihood Principle can now be applied to actually correct the error and obtain the original data (Premkumar, et. al. 2002). Additional combinations of nonredundant and redundant residues are formed. Each combination can then be converted using the Chinese Remainder Theorem. The value within the legitimate range that appears the most often will be the correct value of the original data. Continuing with the previous example where residue r1 for the decimal number 220 was corrupted into the erroneous value of 3, the additional residue combinations of $\{r_1, r_5, r_6\} = \{3, 0, 12\}$, $\{r_2, r_5, r_6\} = \{3, 0, 12\}$, $\{r_3, r_5, r_6\} = \{4, 0, 12\}$ can be created. Conversion using the Chinese Remainder Theorem yields the values X = 363, X = 220, X = 220 for the residue sets $\{r_1, r_5, r_6\}$, $\{r_2, r_5, r_6\}$, $\{r_3, r_5, r_6\}$ respectively. As 220 appears the greatest number of times it is known to be the correct original data.

RESULTS

A custom processor architecture designed specifically for pipelining and parallelizing the calculations required for this error detection and correction technique has been designed and implemented in a Field Programmable Gate Array (FPGA). The specific moduli set implemented was the set {5, 7, 8, 9, 11, 13} that was used in this paper's discussion. Testing of the chip on various data sets with and without errors has demonstrated that it does indeed detect and correct errors that exist in received residue sets. Only four clock cycles are needed by the chip to detect and correct an error.

CONCLUSIONS

A pipelined and parallelized computer chip has been developed that allows for the automatic detection and correction of corrupted data using redundant residue number systems and the Maximum Likelihood Principle. All testing of the design indicates that the chip's computations match the theoretically predicted results. Moreover, if data has been corrupted, the error can be corrected in only four clock cycles helping to increase the transmission rate of a communication system that uses this design.

LITERATURE CITED

Premkumar, A.B., A.S. Madhukumar, and C.T. Lau 2002. Applying Maximum Likelihood Principles for Error Correction in Residue Number Domain. IEEE Trans. On Circuits and Systems. 2:135-144.

Szabo, N.S., and R.I. Tanaka. 1967. Residue Arithmetic and its Application to Computer Technology. McGraw Hill. 236 pp.