The Fast Algorithm For Number Comparing In Three-Modular RNS

Nikolay Chervyakov
Department of Mathematics and Mathematical Modelling.
North-Caucasus Federal University
Stavropol, Russian Federation,
k-fmf-primath@stavsu.ru

Anton Nazarov,
Department of Mathematics and Mathematical Modelling.
North-Caucasus Federal University
Stavropol, Russian Federation,
kapitoshking@mail.ru

Mikhail Babenko
Department of Mathematics and Mathematical Modelling.
North-Caucasus Federal University
Stavropol, Russian Federation,
mgbabenko@ncfu.ru

Anastasiia Garianina
Department of Mathematics and Mathematical Modelling.
North-Caucasus Federal University
Stavropol, Russian Federation,
garyanina.anastasia@gmail.com

Andrei Tchernykh
CICESE Research Center,
Ensenada, Baja California, Mexico
chernykh@cicese.mx

***Abstract*— In paper the parallel algorithm of number comparing in residue number system of the special form (2^n-1, 2^n, 2^n+1) based on use of diagonal functions from work [5] and a method of the recursive doubling from work [9] is offered. The offered algorithm allows scoring 31% in area of the occupied device and 18% in time delay in comparing with algorithm from work [9].**

Keywords- residue number system; diagonal functions; Chinese Remainder Theorem

I. INTRODUCTION

Residue Number System (RNS) is non-positional number system that allows to implement a big class of tasks [1]. Each digit in RNS is the remainder of division on some number, called the base. All the bases for each digit make a system of the RNS bases. The uniqueness of RNS number representation is guaranteed only in case of that all system bases are pairwise co-primes. Let (m_1, m_2,...,m_n) – given system of bases, uniquely defines some RNS. In this RNS $X \xrightarrow{RNS} (x_1, x_2, ..., x_n)$, where $x_i = |X|_{m_i}$ for $i = \overline{1,n}$. At the same time X should be in an interval [*0, M*), where $M = \prod_{i=1}^{n} m_i$ is called the range of given RNS. The main RNS advantages are in the opportunities of parallel representation and data handling in case of use the linear systems algorithms - the algorithmic structures requiring execution generally of addition and multiplication operations by analogy with the linear algebraic operators - as a basis. In such cases for RNS inter bit transfers aren't required at the expense of what both data handling time and resource intensity of the used algorithms is reduced.

However, it makes the comparison of numbers more difficult than in the traditional weighted number systems. Efficient methods for number comparison in RNS is an important topic of research in the last few years. We address number comparing of a special type (2^n-1. 2^n. 2^n+1).

Such operations require executions of addition operations [2], computation of the positional characteristic on the basis of the Chinese Remainder Theorem [3,4], diagonal functions [5], recursive doubling methods [6], approximate methods [7, 8], and others.

II. COMPARING ON THE BASE OF DIAGONAL FUNCTION

It is known that there is a simple algorithm of number comparing in a weighted number system on some base p. Execution of this algorithm comes down to sequential comparing of digits of the appropriate bits of representation. In not positional number systems, for example in residue number system, there is no such a simple algorithm. All number comparing algorithms in such systems come down, in essence, to preliminary conversion of not positional representation to some positional, and then – to usual number comparing in weighted number system [4]. Thus, these algorithms, if it is possible to tell so, are external in relation to the number representation system.

In paper [5] the method, allowing to compare two numbers, based on use of diagonal functions. The diagonal function is set by a formula:

978-1-5090-4553-2/16 $31.00 © 2016 IEEE
DOI 10.1109/EnT.2016.12

$$D(X) = \sum_{i=1}^{n} \left| \frac{X}{m_i} \right|$$

For convenience of computation of diagonal function value they use the following formula:

$$D(X) = \left| \sum_{i=1}^{n} k_i x_i \right|_{SQ} \quad (1)$$

where $SQ = \prod_{i=1}^{n} M_i$, $k_i = \left| -\frac{A}{m_i} \right|_{SQ}$ and $x_i = |X|_{m_i}$ for all $i=1,2, \ldots, n$.

Algorithm 1. Number Comparison in the Residue Number System

Input: $X \xrightarrow{RNS} (x_1, x_2, \ldots, x_n)$, $Y \xrightarrow{RNS} (y_1, y_2, \ldots, y_n)$.

Output: "10" if $X>Y$; "00" if $X=Y$; "01" if $X<Y$.

1. Computing $D(X)$ and $D(Y)$ by Eq.(1);
2. If $D(X)> D(Y)$ then return"10";
3. If $D(X)< D(Y)$ then return "01";
4. If $D(X)>= D(Y)$ then
4.1. If $x_1>y_1$ then return"10";
4.2. If $x_1<y_1$ then return"01";
4.3. If $x_1=y_1$ then return"00";

As the diagonal function is not strictly monotonic, then for number comparing in general it requires additional comparing. However, for RNS modules of a special form (2^n-1, 2^n+1) the diagonal function allows to avoid operation of taking a remainder of division on the module 2^{2n}-1 and to replace it with the one of finding a remainder of division on the module 2^{n+1}, that is equivalent to operation of taking of low n+1 bits of number, that allows to increase the speed of execution of arithmetical operations of number comparing in RNS with modules (2^n-1, 2^n+1).

III. Comparing of Numbers in RNS of Special Form

Let an integer positive number X be a set in RNS (x_1,x_2,x_3), where $x_1 = |X|_{2^n-1}$, $x_2 = |X|_{2^n}$, $x_3 = |X|_{2^n+1}$. Using the approach from work [6], we can represent X in the following way:

$$X = x_2 + |k(x_1 - x_2)|_{2^n-1} 2^n \quad (2)$$

where $k = \left| \frac{A}{2^n} \right|_{2^n-1} = 2^n$, X_1 binary representation of number (x_1, x_3) in RNS $(2^n-1, 2^n+1)$. Let $f(X) = |k(x_1 - x_2)|_{2^n-1}$, then $|f(X)|_{2^n-1} = |x_1 - x_2|_{2^n-1}$ and $|f(X)|_{2^n+1} = |x_2 - x_3|_{2^n+1}$.

For RNS $(2^n-1, 2^n+1)$, the parameters of diagonal function are: $SQ=2^{n+1}$, $k_1 = \left| -\frac{A}{2^n-1} \right|_{2^{n+1}} = 2^n + 1$ and $k_2 = \left| -\frac{A}{2^n+1} \right|_{2^{n+1}} = 2^n - 1$.

Diagonal function is the following:

$$\begin{aligned} D(f(X)) = |2^n(&|x_1 - x_2|_{2^n-1} \\ &+ |x_2 - x_3|_{2^n+1}) \\ &+ |x_1 - x_2|_{2^n-1} \\ &- |x_2 - x_3|_{2^n+1}|_{2^{n+1}} \end{aligned} \quad (3)$$

As the expression $|2^n(|x_1 - x_2|_{2^n-1} + |x_2 - x_3|_{2^n+1})|_{2^{n+1}}$ may have only two values 0 or 2^n, then the formula (3) will be:

$$\begin{aligned} D(f(X)) = |2^n|&(|x_1 - x_2|_{2^n-1} \\ &+ |x_2 - x_3|_{2^n+1})|_2 \\ &+ |x_1 - x_2|_{2^n-1} \\ &- |x_2 - x_3|_{2^n+1}|_{2^{n+1}} \end{aligned} \quad (4)$$

A diagram of the device realizing the formula (4) of computing the value $D(f(x))$ is shown on figure 1.

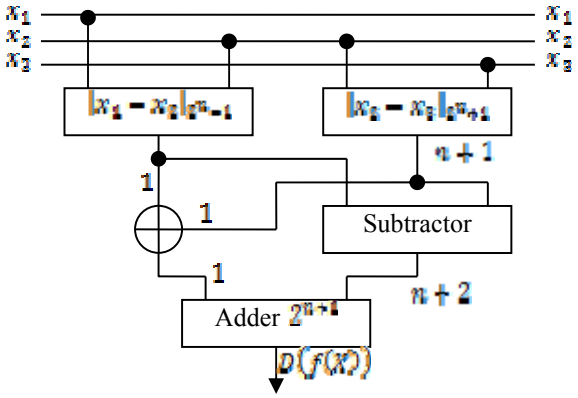

Figure 1. The hardware realization of computing the value D(f(X))

Simultaneous use of the diagonal function and recursive doubling technique avoid costly arithmetic operations. By applying the diagonal function to the set (2^n-1, 2^n+1), we get new effective method of comparing the numbers.

To show practical applicability of the proposed method, we perform a comprehensive study of the practical performance of the proposed algorithm. To this end, we use FPGA Xilinx Virtex xc6vlx760-21760. For the simulation, we take n={4, 8, 12, 16, 20, 24, 28, 32}x`. Results of modeling are shown in Figure 2. We can see that the offered algorithm allows scoring 31% in area of the occupied device and 18% in time delay in comparing with algorithm from work [9].

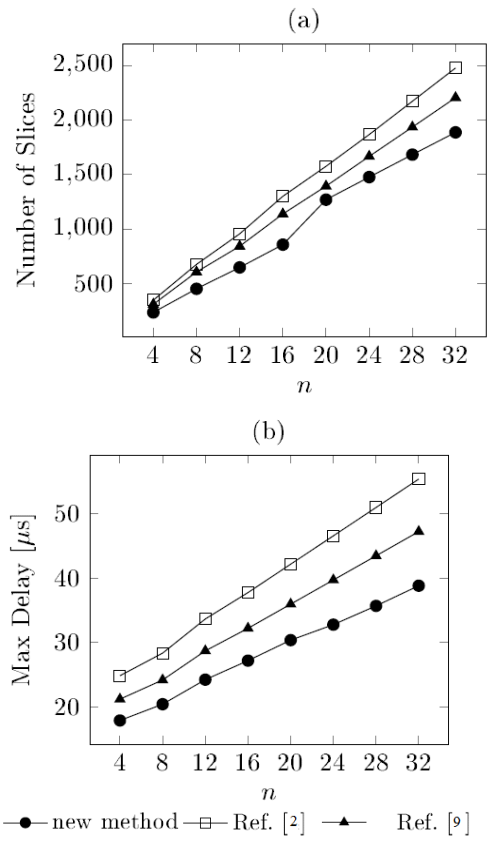

Figure 2. Number of FPGA slices and the maximal delay for various RNS comparators

IV. CONCLUSION

In this paper, we propose an effective algorithm for comparing two integer positive numbers X and Y. We use the formula (2), modified diagonal function (3) from work [5], and compare numbers $f(X)$ and $f(Y)$. In the paper the analysis of comparing methods in three-modulo RNS of special form $(2^n\text{-}1,\ 2^n,\ 2^n\text{+}1)$. Was conducted. The new algorithm based on simultaneous use of algorithm of the recursive doubling offered in work [6] and the diagonal function offered in work [5] is offered. Sharing of these two algorithms allows to leave from computation of residual on the big module equal to RNS range and to pass to computation of transaction on module 2^{n+1} (taking of low bits of number). From results of simulation it is possible to make a conclusion that the offered algorithm allows scoring 31% in area of the occupied device and 18% in time delay in comparing with algorithm from work [9].

ACKNOWLEDGMENT

This work is partially supported by the State task No. 2563 and Russian Federation President Grant SP-1215.2016.5. Part of the work was supported by CONACYT, México, grant no. 178415

REFERENCES

[1] C. H. Chang, A. S. Molahosseini, A. A. E. Zarandi, and T. F. Tay, "Residue Number Systems: A New Paradigm to Datapath Optimization for Low-Power and High-Performance Digital Signal Processing Applications," IEEE Circuits and Systems Magazine, vol. 15, no. 4, 2015, pp. 26-44, doi: 10.1109/MCAS.2015.2484118.

[2] S. T. Eivazi, M. Hosseinzadeh, O. Mirmotahari "Fully parallel comparator for the moduli set," IEICE Electronics Express, vol. 8, no. 12, 2011, pp. 897-901, doi: 10.1587/elex.8.897.

[3] A. Omondi, B. Premkumar, "Residue Number Systems: Theory and Implementation," Imperial College Press, London, UK, 2007.

[4] N.S. Szabo, R.I. Tanaka, "Residue Arithmetic and Its Application to Computer Technology," McGraw-Hill, New York, NY, USA, 1967.

[5] G. Dimauro, S. Impedovo, G. Pirlo, "A new technique for fast number comparison in the residue number system," IEEE Transactions on Computers, vol. 42, no. 5, 1993, pp. 608-612, doi: 10.1109/12.223680.

[6] Y. Wang, X. Song, M. Aboulhamid, "A new algorithm for RNS magnitude comparison based on new Chinese remainder theorem II," In: Proc. ACM Great Lakesc Symp. VLSI (GLSVLSI), 4-6 Mar 1999, pp. 362-365, doi: 10.1109/GLSV.1999.757457.

[7] N. I. Chervyakov, M. G. Babenko, P. A. Lyakhov, and I. N. Lavrinenko, "An approximate method for comparing modular numbers and its application to the division of numbers in residue number systems," Cybernetics and Systems Analysis, vol. 50, no. 6, 2014, pp. 977-984, doi: 10.1007/s10559-014-9689-2.

[8] C. Y. Hung, B. Parhami, "An approximate sign detection method for residue numbers and its application to RNS division," Computers & Mathematics with Applications, vol. 27, no. 4, 1994, pp. 23-35, doi:10.1016/0898-1221(94)90052-3

[9] L. Li, G. Li, Y. Zhao, P. Yin, and W. Zhou, "High Speed Comparator for the Moduli," IEICE Electronics Express, vol. 10, no. 21, 2013, pp. 20130628-20130628, doi: 10.1587/elex.10.20130628