Journal of Computational and Applied Mathematics 309 (2017) 575–586

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Mathematical model and implementation of rational processing

H. Mora *, J. Mora-Pascual, J.M. García-Chamizo, M.T. Signes-Pont

Specialized Processor Architectures Laboratory, Department of Computer Technology, University of Alicante, San Vicente del Raspeig, Alicante (E-03080), Spain

HIGHLIGHTS

- It is introduced a formal framework for processing rational numbers.
- A representation system based on positional notation system is described.
- A method for calculating the addition function is detailed.
- Experiments and application example have been made to validate the model.

ARTICLE INFO

Article history:
Received 10 November 2015
Received in revised form 27 April 2016

Keywords:
Computer arithmetic
Computational methods
Exact computation
Rational processing
Mathematical model
Computational efficiency

ABSTRACT

Precision in computations is a considerable challenge to adequately addressing many current scientific or engineering problems. The way in which the numbers are represented constitutes the first step to compute them and determines the validity of the results. The aim of this research is to provide a formal framework and a set of computational primitives to address high precision problems of mathematical calculation in engineering and numerical simulation. The main contribution of this research is a mathematical model to build an exact arithmetical unit able to represent without error rational numbers in positional notation system. The functions under consideration are addition and multiplication because they form an algebraic commutative ring which contains a multiplicative inverse for every non-zero element. This paper reviews other specialized arithmetic units based on existing formats to show ways to make high precision computing. It is proposed a formal framework of the whole arithmetic architecture in which the operators are based. Then, the design of the addition operation is detailed and its hardware implementation is described. Finally, extensive evaluation of this operator is performed to prove its ability for exact processing.

1. Introduction

1.1. Need for precision in calculations

High precision computing is a very active research area due to the number of interesting applications that need it. For example, research about the nature of matter in the LHC or the search for extrasolar planets requires complex numerical calculations on the edge of the range of representation formats. Without going so far, other much more common everyday

* Corresponding author.
E-mail addresses: hmora@ua.es (H. Mora), jeronimo@ua.es (J. Mora-Pascual), juanma@ua.es (J.M. García-Chamizo), teresa@ua.es (M.T. Signes-Pont).

http://dx.doi.org/10.1016/j.cam.2016.05.001

Table 1
Error representation of decimal numbers coded in simple precision binary floating point format.

	Decimal number	Binary floating point representation	Error
a	0.6	0.600000023841857	2.3E−8
b	0.1	0.100000001490116	1.4E−9
c	0.7	0.699999988079071	1.1E−8
d	0.4	0.400000005960464	5.9E−9
e	0.04	0.039999999105930	1.0E−10
$a + b$	0.7	0.700000025331973	2.5E−8
$a + a$	1.2	1.200000047683715	4.7E−8
$d + d + d$	1.2	1.200000017881392	1.7E−8
$b + c + d$	1.2	1.199999995529651	4.5E−9
$e + \cdots + e$ (10 times)	0.4	0.399999991059300	8.9E−9

operations, such as financial calculations, also require a good precision to avoid inadmissible computing and rounding errors. Table 1 shows some examples of the representation error by floating point binary format codification and operations with simple decimal data. It shows how the same number can have multiple different binary representations and how the accumulative operations increase the error.

The introduction of decimal representation formats has significantly improved the accuracy of applications for processing numerical data introduced by humans through a terminal [1]. The programming languages and database systems include among their types the new data money or decimal to represent 10-base numeric data [2–4]. New processors support these formats and offer a wide instruction set for native execution [5–9]. Although the computational cost is superior to binary, for certain applications, most precision compensates this reduced performance.

However, in other areas the problem of precision remains unresolved since their numeric range of the operands and results are found in the periodic rational or irrational real domain. In these cases, the numbers do not correspond with representable values in binary format nor decimal. For example, next table (Table 2) shows some rational numbers represented in binary and decimal floating point format. These cases make clear the codification and operation error by both standard formats. These systematic errors make it necessary to have a computation model able to represent these numbers and to operate with them without error.

1.2. Challenges and objectives of the work

This work aims to propose a mathematical model to represent and compute rational numbers. This model constitutes the formal framework of an arithmetic architecture where computational techniques are defined to build the operators with rational numbers and perform exact processing.

The key idea of this research is based on representing explicitly the non-zero periodic part of the rational numbers expressed by the positional number system. The challenges of this notation are in developing computational techniques to process the numbers, especially if they are also coded in floating point. So that, this work introduces the calculation method of the addition operator for the proposed rational representation scheme. Its specification for hardware implementation will be detailed in deep. The multiplication function can be designed based on the same principles.

The novelty of this work lies in proposing calculation methods for rational numbers represented in positional number system where periodic numbers can be represented in a direct way without error. The experiments show that this approach is an alternative to the decimal formats for exact coding rational numbers.

This paper is structured as follows: Section 2 provides a review of the current state of knowledge on specialized arithmetic processing. The most relevant proposals and works of this issue are described and some findings about them are stated; Section 3 describes the formal framework and rational functions on which the architectural model is constructed, Section 4 explains the implementation of the rational processing. The overall architecture is introduced, and the representation format and the addition operator are detailed; Section 5 shows an empirical evaluation of the results of processing the double mantissa numbers and an example of application on intensive calculus, finally, Section 6 summarizes the conclusions of this work.

2. Related work

2.1. High precision computing proposals

This section is not intended to contain an exhaustive and detailed review of state-of-the-art, but only introducing the more representative proposals and results that show the progress in the issue of high precision computing.

In first place, the last floating point standard [1] is the most used for number codification and arithmetic computing [10,11]. There are research works which show the limitations of these formats [12] and the error produced by the processing of the floating point arithmetic [13,14].

On the way of search for improving precision in calculations, software solutions are the first stage. There are a great variety of math libraries for numerical calculation with greater precision than conventional standard formats [2–4].