

УДК 004.942

А.Я. Аноприенко, С.В. Иваница, С.В. Кулибаба
Донецкий национальный технический университет
anoprien@cs.dgtu.donetsk.ua, isv@cs.dgtu.donetsk.ua

Представление постбинарных форматов чисел с плавающей запятой в контексте интервальных вычислений

Рассмотрен ряд модификаций традиционных форматов чисел с плавающей запятой. Описаны методы идентификации данных форматов и дано краткое описание каждой модификации. Разработаны алгоритмы определения форматов модифицированных чисел по коду и модификатору формата. Представлена VHDL-модель устройства, выполняющего преобразование традиционных форматов чисел в постбинарные с возможностью их дальнейшего использования как интервальный тип данных в соответствующих (постбинарных, интервальных) вычислениях без потери точности.

Постбинарное число, тетралогика, интервальные вычисления, постбинарные вычисления, плавающая запятая, мантисса

Введение

На сегодняшний день аппаратное обеспечение компьютеров в состоянии производить всевозможные вычисления с числами, которые могут быть представлены всего в двух форматах — целые числа и числа с плавающей запятой. Именно вычисления с плавающей запятой являются основой современного компьютерного моделирования. Для представления действительных чисел (чисел с плавающей запятой) в двоичном коде используется разработанный ассоциацией IEEE (Institute of Electrical and Electronics Engineers) стандарт для двоичной арифметики с плавающей точкой (IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Std 754-1985). В ряде работ [1-3] было отмечено несовершенство стандарта IEEE 754-1985, и на многих примерах показано, как использование стандарта IEEE 754 может привести к множественным ошибкам компьютерных вычислений. В качестве основных ошибок, вызванных применением чисел в формате IEEE 754-1985 можно назвать ошибки связанные с точностью представления вещественных чисел, с неправильным приведением типов данных, а также вызванные сдвигом мантисс и округлением [1]. В 2008 году ассоциация IEEE выпустила расширенный стандарт IEEE 754-2008, который без изменения включил в себя стандарт 1985 года и дополнил его форматом учетверенной точности. Но введение нового формата не позволило решить основные проблемы, связанные с появлением грубых ошибок в вычислениях [3], что является главной предпосылкой для существенной модификации всей системы компьютерных вычислений с целью повышения ее надежности и адаптации к современным требованиям, особенно с точки зрения компьютерного моделирования.

Целью данной работы является представление постбинарных форматов чисел с плавающей запятой для хранения как одного вещественного числа (аналогично представлению числа в форматах стандарта IEEE 754), так и пары чисел, являющихся либо числителем и знаменателем дроби (дробный формат) либо границами интервала (интервальный формат). Приведены блок-схемы определения формата числа по значению кода формата и выборки полей мантиссы, порядка и знака по значению модификатора формата. В рамках практической реализации постбинарных форматов представлен преобразователь, который осуществляет переход от вещественного формата к интервальному с возможностью хранения полученного интервального значения в одном поле постбинарного формата числа.

Постбинарное представление вещественных чисел

В современных условиях в качестве одного из решений проблем, связанных с форматом представления чисел с плавающей запятой может выступать дальнейшее развитие как логической, так и вычислительной составляющей современного компьютеринга [4]. В работе [3] были предложены способы для преодоления проблем, связанных с ограничением разрядности чисел, поскольку использование при вычислениях разрядности, существенно превышающей стандартную, приводит к получению правильных результатов. К таким способам можно отнести операции, приводящие к увеличению (или выравниванию) разрядности во избежание переполнения порядка результата и выполнения корректных вычислений, выполнение отложенного деления (числитель и знаменатель вычисляются отдельно, а деление откладывается

на заключительный этап вычислений), а также использование интервальных вычислений [5, 6].

Реализация подобных операций была заложена в разработку постбинарных методов вычислений, основанных на постбинарном представлении количественных значений [4, 7].

Поэтому на основании форматов чисел различной точности стандарта IEEE754-2008 был предложен ряд постбинарных форматов, общая схема полей которых представлена на рис. 1, а размерность полей — в табл. 1.

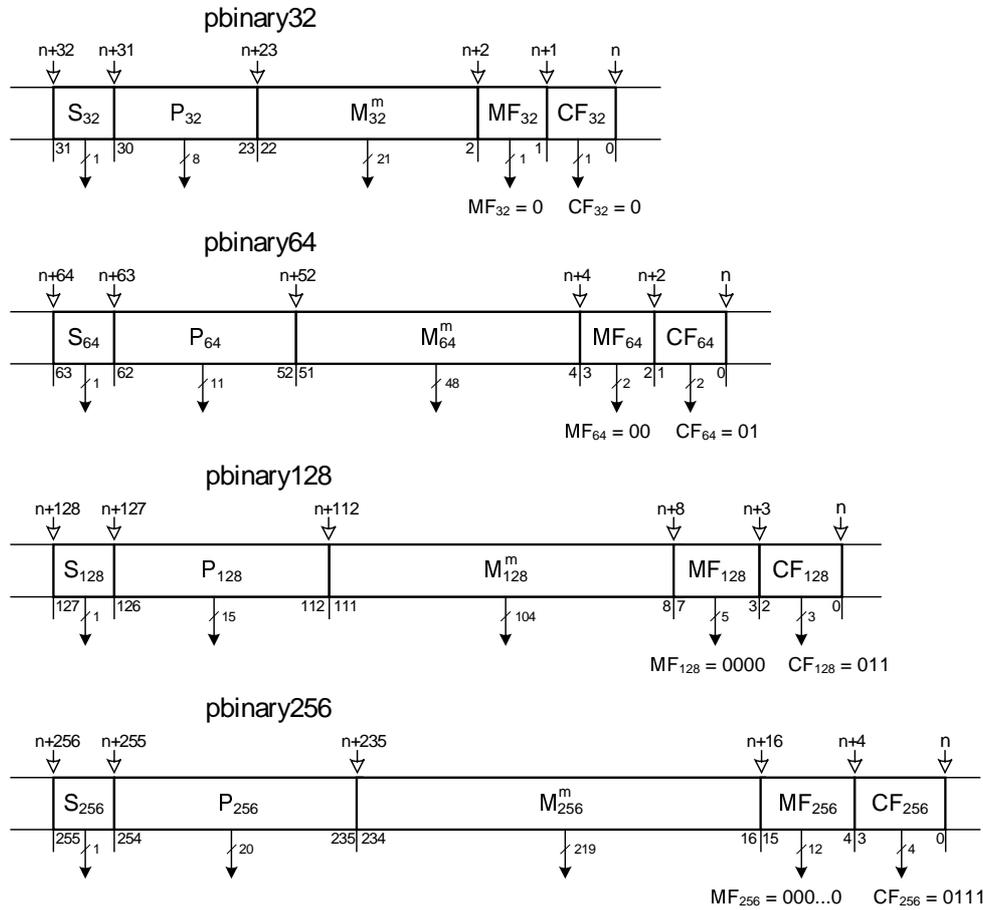


Рисунок 1 — Предлагаемые варианты модифицированных форматов чисел с плавающей запятой (S, E, M^m — знак, порядок и мантисса числа, MF, CF — модификатор и формат числа, n — указатель на младший бит формата)

Таблица 1 — Разрядность полей постбинарных форматов чисел (в скобках возле количества разрядов модифицированной мантиссы M^m приведено количество разрядов традиционной мантиссы M числа соответствующей точности)

Формат	Количество двоичных разрядов полей формата					Значение поля CF
	S	E	M ^m (M)	MF	CF	
pbinary32	1	8	21 (23)	1	1	0
pbinary64	1	11	46 (52)	2	2	01
pbinary128	1	15	104 (112)	5	3	011
pbinary256	1	20	225 (235)	12	4	0111

Для обеспечения гибкого использования различных форматов чисел с плавающей запятой вводится поле идентификатора формата, для которого в стандартных форматах могут быть

задействованы часть младших разрядов мантиссы M и которое в общем случае состоит из кода (определителя формата числа) CF и модификатора MF формата. Мантисса, уступившая часть

младших разрядов для значений полей CF и MF становится модифицированной M^m . По значениям поля MF формат числа уточняется и может быть представлен как «парный формат», в котором путем удвоения каждого из разрядов в виде единого «постбинарного значения» представляются пары чисел, интерпретируемые либо как числитель и знаменатель обыкновенной дроби, либо как граничные значения интервалов (для поддержки интервальных вычислений) [3].

Поскольку значение кода формата CF сформировано по принципу унитарного кодирования, то алгоритм определения размерности числа сводится к определению положения первого нуля, начиная с младших разрядов поля числа (например, начиная со значения по указателю n, рис. 1). На рис. 2 приведена блок-схема алгоритма определения формата числа по значению кода формата.

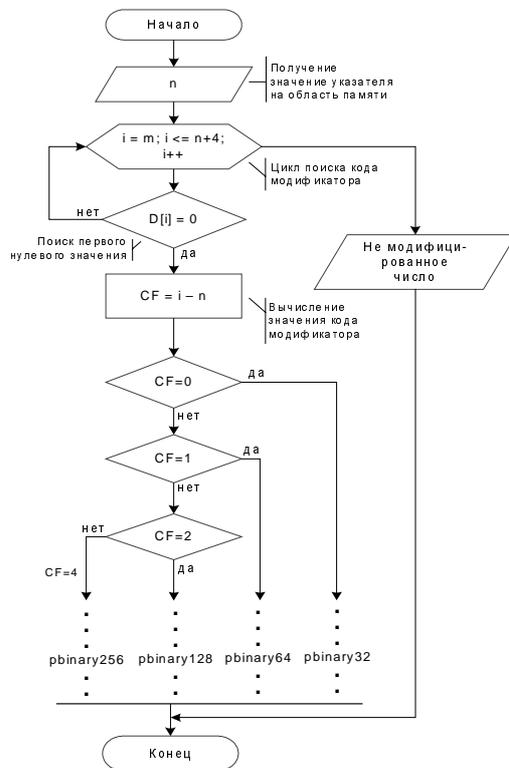


Рисунок 2 — Блок-схема определения формата числа по значению CF

После определения общего формата числа могут быть получены значения указателя на поля модификатора формата, мантиссу, порядок и знак числа. Например, для формата pbinary32, где n — указатель на младший бит формата (см. рис. 1), по указателю n + 1 происходит выборка бита модификатора формата MF. По значению MF определяется тип находящегося в данном формате числа. При MF = 0 в поле формата находится вещественное число, причём

n + 2 — указатель на младший бит мантиссы числа; n + 23 — указатель на младший бит порядка числа; n + 31 — указатель на знаковый бит числа. Типы чисел в зависимости от значения MF для рассматриваемых постбинарных форматов представлены в табл. 2–5.

Таблица 2 — Модификации 32-разрядного постбинарного формата

MF	Модификация pbinary32
0	pbinary32
1	pbinary32/16p

Таблица 3 — Модификации 64-разрядного постбинарного формата

MF	Модификация pbinary64
00	pbinary64
01	pbinary64/32f
10	pbinary64/32i
11	pbinary64/32p

Таблица 4 — Модификации 128-разрядного постбинарного формата

MF	Модификация pbinary128
00000	pbinary128
00001	pbinary128/64f
00010	pbinary128/64i
00011	pbinary128/64p
00100	pbinary128/32fp
00101	pbinary128/32ip
00101 ... 11111	резерв

Таблица 5 — Модификации 256-разрядного постбинарного формата

MF	Модификация pbinary256
000000000000	pbinary256
000000000001	pbinary256/128f
000000000010	pbinary256/128i
000000000011	pbinary256/128p
000000000100	pbinary256/64fp
000000000100	pbinary256/64ip
000000000101	резерв
... 111111111111	

На примере формата pbinary128 рассмотрим структуру модифицированных форматов:

- pbinary128 — формат вещественного числа четвертичной точности (рис. 1);
- pbinary128/64f — дробный формат числа (индекс f): в 128-разрядном поле находятся два 64-разрядных числа, являющиеся числителем и знаменателем дроби;

- `rbinary128/64i` — інтервальний формат числа (індекс `i`): в 128-разрядному полі знаходяться два 64-разрядних числа, являючися границями інтервала;

- `rbinary128/64p` — постбінарне число (індекс `p`): в 128-разрядному полі знаходиться 64-разрядне число, представлене значеннями тетракода 0, 1, A, M [8], кожне з яких кодується парой двоичних разрядів;

- `rbinary128/32fp` — дробний постбінарний формат числа (індекс `fp`): представляє сукупність форматів `rbinary128/64i` і `rbinary64/32f`, т. е. 128-разрядне поле числа розділено на два 64-разрядних поля, в яких знаходяться 32-разрядні значення тетракодів, представляючі чисельник і знаменатель дробу;

- `rbinary128/32ip` — інтервальний постбінарний формат числа (індекс `ip`): представляє сукупність форматів `rbinary128/64i` і `rbinary64/32p`, т. е. 128-разрядне поле числа розділено на два 64-разрядних поля, в яких знаходяться 32-разрядні значення тетракодів, представляючі границі інтервала.

Разработка преобразователя постбинарных чисел

На основани разработанных форматов представления вещественных чисел средствами языка VHDL [9] описана модель устройства, выполняющего преобразование чисел в условный постбинарный тип данных (одно значение сформированного тетракода условно кодируется одним двоичным разрядом) с дальнейшим использованием полученного формата в интервальных вычислениях. Фактически разработан алгоритм преобразования одного постбинарного числа, которое, в конечном счете, представляет собой значения границ интервала. Структура модели представлена на рис. 3.

Рассмотрим работу модели. Пусть на входную 32-разрядную шину `inDATA` поступило, например, двоичное значение десятичной дроби 0,123456789. Однако данное число не может быть точно представлено в формате одинарной точности (`Single`), соответствующего стандарту IEEE754-2008, и после обратного преобразования в десятичный формат, в окне Windows будет возвращено число 0,1234568 (по результатам работы программы IEEE754 v.1.00 [10]), а действительное представление полученного числа (возвращенное из `Single`) будет следующим:

+0,12345679104328155517578125.

Полученное число несет в себе ошибку точности представления вещественного числа формата `Single`. Во избежание такого рода ошибок, в блоке `dt_converter` происходит преобразование бинарного числа в постбинарное,

в котором каждый бит заменяется значением тетракода, отображающего одно из четырех состояний тетралогии: логические ноль и единица, состояния множественности (многозначности) и неопределенности [4]. Причем значения множественности заносятся в те разряды мантиссы, начиная с которых возникает возможная ошибка округления (номер бита, с которого начинаются разряды множественности, и количество таких разрядов поступают на соответствующие входы `inNumber` и `inCount` блока `dt_converter`). Младшие разряды, начиная от последнего разряда множественности, заполняются значениями неопределенности, поскольку уже не являются значащими. Так, двоичное представление числа 0,123456789 в 32-разрядном формате с плавающей запятой выглядит следующим образом (значение на шине `inDATA`):

0 01111011 11111001101011011101010.

После обработки в блоке `dt_converter` на шину `outDT` поступает постбинарное число, которое имеет следующий вид:

0 01111011 11111001101011011MMAAAA.

В блоке `TI_converter` происходит разбиение полученного постбинарного числа на два бинарных, представляющих собой границы интервального результирующего числа. Процедура перехода постбинарного числа в интервальное происходит по принципу, описанному в [7]. При этом все разряды множественности заменяются либо двоичным «0» для формирования левой (нижней) границы интервала, либо двоичной «1» для формирования правой (верхней) границы интервала. Значения неопределенности для каждой границы заполняются случайными двоичными значениями, которые формируются в соответствующих генераторах псевдослучайных чисел. Полученные значения границ интервала поступают на соответствующие 32-разрядные выходные шины `outLow` и `outHigh`. Блок `ALU` разработан для реализации логических и арифметических операций над постбинарными числами.

На рис. 4 представлена временная диаграмма работы модели данного устройства, из которой видны все промежуточные значения и значения результирующего интервала (подчеркнутые поля чисел могут быть заполнены случайным набором двоичных значений):

[0 01111011 11111001101011011001111;

0 01111011 1111100110101101111011].

Преобразовав данный интервал в 10 с/с, получим следующую пару значений:

[0,1234566; 0,1234569],

представляющих собой границы интервала, имеющего достаточно узкую ширину $wid = 2,458692e-07$.

При этом выполняется следующее условие: $0,123456789 \in [0,1234566; 0,1234569]$,

свидетельствующее о том, что полученный интервал представляет исходное число, являющееся элементом множества R действительных чисел, на множестве IR интервальных чисел.

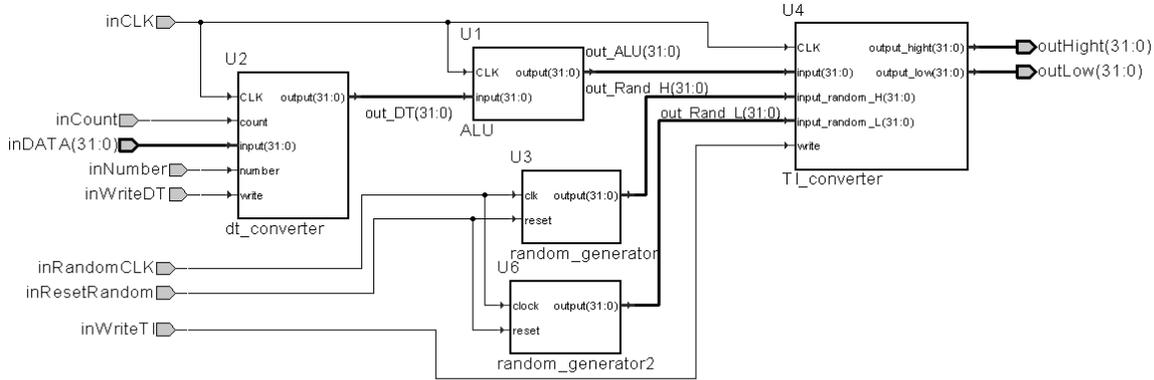


Рисунок 3 — Схема преобразователя (dt_converter — блок преобразования входного вещественного числа в постбинарное, TI_converter — блок разложения постбинарного числа на границы интервала, ALU — арифметико-логическое устройство; random_generator — генератор псевдослучайных чисел)

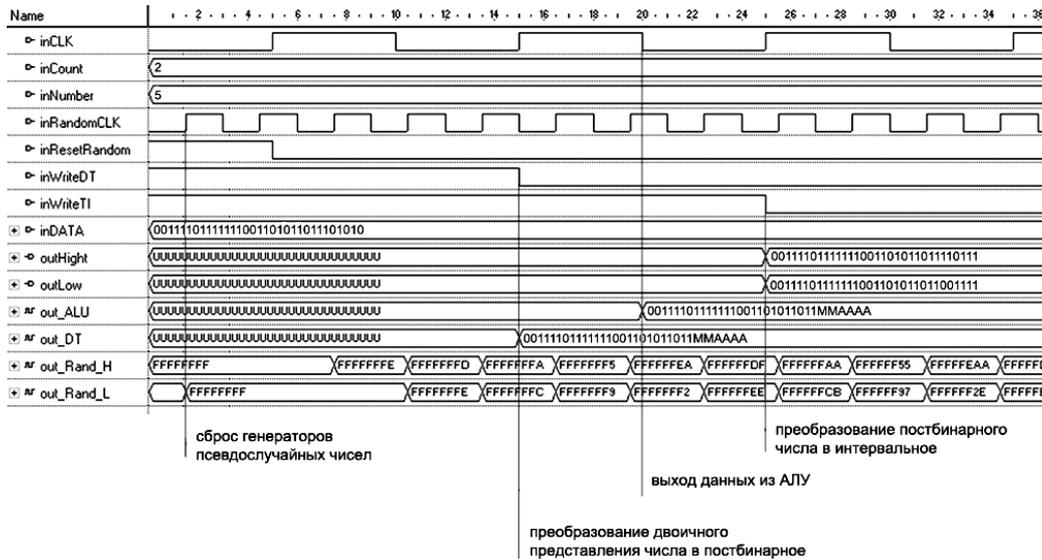


Рисунок 4 — Временная диаграмма работы преобразователя

Заключение

В данной работе предложен последовательный переход к различным представлениям числа (от десятичного к интервальному через постбинарное) и модель соответствующего преобразователя, на вход которого поступают десятичные числа. Все внутренние вычислительные операции выполняются с постбинарными числами, а результатом является интервальное число, гарантировано включающее в себя точное значение выполненных операций.

Для устранения необратимых погрешностей, возникающих в ходе вычисления и приводящих к неверному результату, предложены модификации стандарта форматов представления чисел от половинной до октоточности.

В ходе работы модели вычислительное устройство в качестве преобразователя чисел показало высокую точность работы. В дальнейшем планируется разработка алгоритмов работы с вещественными тетрадовыми и интервальными числами, которые будут составлять основу построения постбинарного

процессора. Также планируется построение программной модели вычислителя, в которой, во-первых, все значения в зависимости от условия вычислений будут представлены в различных модификациях постбинарных форматов, во-

вторых, будет обеспечена необходимая точность полученных результатов путем реализации, как механизма гибкой разрядности, так и перехода к вычислениям с дробями и интервалами.

Список литературы

1. Яшкардин В.Л. IEEE 754: стандарт двоичной арифметики с плавающей точкой: В.Л. Яшкардин. – [Электронный ресурс]. – Страница доступа: <http://www.softelectro.ru/ieee754.html>.
2. Юровицкий В.М. IEEE754-тика угрожает человечеству / В.М. Юровицкий. – [Электронный ресурс]. – Страница доступа: <http://www.yur.ru/science/computer/IEEE754.htm>
3. Аноприенко А.Я. Пример Румпа в контексте традиционных, интервальных и постбинарных вычислений / А.Я. Аноприенко, В.А. Гранковский, С.В. Иваница // Научные труды Донецкого национального технического университета. Серия: Проблемы моделирования и автоматизации проектирования динамических систем (МАП-2011). – 2011. – Вып. 9 (179). – С. 324–343.
4. Аноприенко А.Я. Обобщенный кодо-логический базис в вычислительном моделировании и представлении знаний: эволюция идеи и перспективы развития / А.Я. Аноприенко // Научные труды Донецкого национального технического университета. Серия: Информатика, кибернетика и вычислительная техника (ИКВТ-2005). — 2005. — Вып. 93. – С. 289-316.
5. Moore R.E. Interval analysis. Eiiglewood Cliffs / R.E. Moore – N.J.:Prentic-e-Hall, 1966.
6. Аноприенко А.Я. Интервальные вычисления и перспективы их развития в контексте кодо-логической эволюции / А.Я. Аноприенко, С.В. Иваница // Научные труды Донецкого национального технического университета. Серия: Проблемы моделирования и автоматизации проектирования динамических систем (МАП-2010). – 2010. – Вып. 8 (168). – С. 150–160.
7. Аноприенко А.Я. Особенности постбинарного кодирования на примере интервального представления результатов вычислений по формуле Бэйли-Боруэйна-Плаффа / А.Я. Аноприенко, С.В. Иваница // Научные труды Донецкого национального технического университета. Серия: Информатика, кибернетика и вычислительная техника (ИКВТ-2010). — 2010. – Вып. 11 (164). – С. 19-23.
8. Аноприенко А.Я. Тетралогия и тетракоды / А.Я. Аноприенко // Сборник трудов факультета вычислительной техники и информатики. – 1996. – Вып.1.
9. Суворова Е. Проектирование цифровых систем на VHDL / Е. Суворова, Ю. Шейнин. – С-Пб.: ВHV, 2003. — С. 576.
10. Бесплатные программы SoftElectro: программа IEEE754 - конвертор чисел формата IEEE754 с абсолютной точностью представления результата. – Страница доступа: <http://www.softelectro.ru/program.html>.

Надійшла до редколегії 18.10.2011

**О.Я. АНОПРИЄНКО, С.В. ІВАНИЦЯ,
С.В. КУЛІБАБА**

Донецький національний технічний університет
Представлення постбінарних форматів чисел з рухомою комою в контексті інтервальних обчислень

Розглянуто ряд модифікацій традиційних форматів чисел з рухомою комою. Описано методи ідентифікації даних форматів і розміщено короткий опис кожної модифікації. Розроблено алгоритми визначення форматів модифікованих чисел за кодом і модифікатором формату. Представлена VHDL-модель пристрою, що виконує перетворення традиційних форматів чисел в постбінарне з можливістю їх подальшого використання як інтервальний тип даних у постбінарних (інтервальних) обчисленнях без втрати точності.

постбінарне число, тетралогіка, інтервальні обчислення, постбінарні обчислення, рухома кома, мантиса, експонента

**A.Y. ANOPRIENKO, S.V. IVANITSA,
S.V. KULIBABA**

Donetsk National Technical University
Representation of Postbinary Floating Point Numbers for Interval Computing

We considered a number of modifications of the traditional formats of floating point numbers. The paper describes the methods of identifying formats and provides description of each modification. The algorithms to determine the formats of modified numbers by the code and format modifier are developed. A VHDL-model of the device performing conversion of traditional formats of numbers into postbinary with possibility of their further use as an interval data type in postbinary (interval) calculations without loss of accuracy is presented.

postbinary number, tetralogic, interval computing, postbinary calculating, floating point, mantissa, exponent