

**Брусенцов Н.П., Владимирова Ю.С.,
Рамиль Альварес Х.**

Троичный логико-алгебраический и арифметический процессор

Существенные достоинства троичного арифметического процессора были выявлены К.Шенноном в статье о симметричных кодах чисел [1], на которую почему-то практически отсутствуют ссылки. Затем, двадцать лет спустя, Д.Кнут охарактеризовал уравновешенную троичную систему счисления как “быть может, самую изящную”, предвещая ей серьезные применения, когда “флип-флоп” заменится на “флип-флэп-флоп” [2, с.218]. Надо сказать, что за десять лет до этого прогноза уже существовала и была выпущена небольшой серией наша машина “Сетунь”, на практике подтвердившая преимущества симметричной троичности. А теперь установлено, что еще в 1840 г. Томас Фаулер сконструировал и построил работающий в симметричном троичном коде механический вычислитель [3]. Странно, что все еще сохраняется представление, будто главное достоинство троичного кода в том, что он самый экономный, поскольку 3 - ближайшее к основанию натуральных логарифмов целое число. Ведь экономия относительно двоичного кода теоретически менее 5% и едва ли реализуема технически. Вместе с тем то, что троичный код является простейшим, позволяющим осуществить симметричную (уравновешенную, сбалансированную) систему счисления, оказывается исключительно важным как в принципе, так и практически. Двоичная система счисления, не обеспечивающая непосредственного представления числа со знаком, удовлетворительного округления и варьирования длины операндов, по сравнению с симметричной троичной явно неполноценна. Пример “Сетуни” наглядно показал, что троично-симметричный арифметический процессор естественней, проще, дешевле и эффективней двоичных.

Но в несравнимо большей степени неадекватность двоичности проявилась при исследовании логико-алгебраических основ информатики [4]. Установлено, что так называемые парадоксы материальной импликации, устранить которые безуспешно пытались виднейшие логики, обусловлены противоестественным “законом исключенного третьего” и составляют неотъемлемую особенность двухзначности. Полноценное содержательное следование, представленное в аристотелевой силлогистике общеутвердительной посылкой “*Всякое x есть y* ” (“Из x с необходимостью следует y ”), в двухзначной логике непредставимо и вырождено в утратившую смысловую взаимосвязь терминов импликацию, что и обусловило бессодержательность этой логики. Основопо-

ложники современной математической логики Д.Гильберт и В.Аккерман отклонились от Аристотеля в истолковании общеутвердительной посылки, оправдывая это “потребностями математических применений логики, где класть в основу аристотелево понимание было бы нецелесообразно” [5, с.79]. На самом деле их отклонение было предопределено двухзначностью, введенной в логику античными стоиками, тогда как подлинная силлогистика трехзначна и содержательна. В условиях двухзначности, наряду с парадоксами материальной импликации, частные посылки не подчинены общим, т.е. отвергается силлогизм подчинения: “*Всякое x есть y*” не влечет “*Некоторые x есть y*”, и поэтому вопреки здравому смыслу отклонены вполне очевидные модусы в 3-й и 4-й фигурах. Понятно, почему двухзначная “классическая” (а в сущности ущербная) логика не используется при решении реальных проблем и не прижилась в школе, хотя предмета, призванного систематически развивать умы, там явно не хватает. Впрочем, теперь пробел устранен - логика покоряет школу, обретая высокотехнологичный облик компьютерной информатики. Но компьютеры двоичные, логика все та же двухзначная, последствия компьютеризации пагубны [6], причем не только в системе образования.

Реальная возможность выхода из сложившейся катастрофической ситуации - внедрить адекватный троичный компьютер с благоразумной содержательной логикой и безупречной уравновешенной арифметикой. Теоретически все еще далеко не всеми осознаваемые достоинства симметричной троичности, будучи овеществленными в “Сетуни”, с завидной легкостью осваивались и успешно применялись каждым, кому довелось работать с этой машиной. По-своему “правильно” поняли суть дела даже бюрократы, добившиеся прекращения ее выпуска и дальнейших разработок троичной цифровой техники. Сегодня троичный компьютер, укомплектованный полноценным логико-алгебраическим процессором, станет действенным средством воссоздания аристотелевой содержательной логики, первоосновы троичной диалектической информатики.

Ключевой элемент воссоздаваемой адекватной логики - отношение содержательного, необходимого следования. Именно к нему стремились устранители парадоксов материальной импликации и изобретатели трехзначных импликаций, также не достигшие цели, поскольку действовали формально, не вникая в содержание того, к чему стремятся. А материальная импликация естественно и просто трансформируется в содержательное следование, если принять во внимание, что в ее характеристической функции

$$\rightarrow(x, y) = xy \vee x'y \vee x'y'$$

статус члена $x'y$ не равносильен статусу членов xy и $x'y'$ [7, 8]. Класс вещей, удовлетворяющих отношению $x \Rightarrow y$ (“из x следует y ”, “ y содержит-

ся в x''), необходимо включает xu -вещи и $x'u'$ -вещи, не может включать xu' -вещей, а включенность $x'u$ -вещей несущественна, приводяща - могут быть, могут не быть. Обозначив значение истинности приводящего буквой σ , $0 < \sigma < 1$, можно охарактеризовать необходимое содержательное следование $x \Rightarrow y$ и соответствующий нечеткий класс трехзначной функцией

$$\Rightarrow(x, y) = xy \vee \sigma x'u \vee x'u'$$

В троичном компьютере эта функция и охарактеризованное ею отношение кодируются значением упорядоченной четверки тритов (ДК-шкалой) $+ - 0 +$, тогда как материальной импликации $x \rightarrow y$ соответствует $+ - ++$. Произвольное n -терминное отношение кодируется 2^n -тритной шкалой, позволяющей отобразить и всякое отношение меньшей арности. Но содержательной логике принадлежат только трехзначные отношения, удовлетворяющие диалектическому принципу сосуществования противоположностей [9]. Над ДК-шкалами определены потритные операции конъюнкции, дизъюнкции и инверсии, а также отображения данной шкалы в шкалу большей арности и извлечения следующих из нее шкал меньшей арности. Этот набор операций, представляющих собой обобщение булевой и теоретико-множественной алгебр, оказался достаточным для исчерпывающей алгебраизации и компьютеризации силлогистики [10].

Арифметический сопроцессор наследует двухстековую (стек операндов и стек процедур) архитектуру "Сетуни 70", модифицированную соответственно требованиям структурированного программирования Э.Дейкстры и пополненную командой exit выхода из подпрограммы. Посредством этой команды обеспечена структурируемость всех неструктурирующихся традиционными средствами алгоритмов [11, с.82-86].

Литература

1. Shannon C.E. A symmetrical notation for numbers. // The American Mathematical Monthly, v. 57, n 2, Feb. 1950, pp. 90 - 93
2. Кнут Д. Искусство программирования для ЭВМ. Получисленные алгоритмы. Т.2. м.: Мир, 1977.
3. The ternary calculating machine of Thomas Fowler. / M.Glusker, D.M.Hogan, P.Vass. // IEEE Annals, v. 27, n. 3, July-Sept. 2005, pp. 4-22.

4. Брусенцов Н.П. Неадекватность двоичной информатики. // Современные информационные технологии и ИТ-образование». Сб. докладов. – М.: МАКС Пресс, 2005. С. 501-503.

5. Гильберт Д., Аккерман В. Основы теоретической логики. – М: ИЛ, 1947.

6. Компьютеры и обучение. / Н.П.Брусенцов, Ю.С.Владимилова, Х.Рамиль Альварес // Вестник Моск. ун-та. Сер. Педагогическое образование. 2005, №1. С. 103 - 105.

7. Брусенцов Н.П. О сложности и запутанности проблемы логического следования. // Современная логика: проблемы теории, истории и применения в науке. Материалы VIII общероссийской научной конференции. 24-26 июня 2004 СПб, 2004. С. 231-233.

8. Брусенцов Н.П., Владимиров Ю.С. Булевы уравнения и логический вывод. // Программные системы и инструменты № 5. Под ред. Л.Н. Королева.– М.: Изд. отд. ф-та ВМиК МГУ, 2005. С. 10-12.

9. Брусенцов Н.П. Интеллект и диалектическая триада. // Искусственный интеллект, 2'2002. – Донецк, 2002. С. 53-57.

10. Брусенцов Н.П. Реанимация аристотелевой силлогистики. // Реставрация логики. – М.: Фонд «Новое тысячелетие» – 2005. С. 140-145.

11. Брусенцов Н.П. Микрокомпьютеры. – М.: Наука 1985.

Доложено на Ломоносовских чтениях на факультете ВМиК МГУ 19 апреля 2005 г.

Опубликовано: Программные системы и инструменты № 6. Под ред. Л.Н. Королева.– М.: Издательский отдел ВМиК МГУ, 2005. С. 184-187.