



## НЕКОТОРЫЕ ВОПРОСЫ ПОВЫШЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ СИГНАЛЬНЫХ ПРОЦЕССОРОВ

Александр Палагин, Мирослав Семотюк, Ярослав Визор, Евгений Чичирин

Институт кибернетики им. В.М. Глушкова  
Национальной академии наук Украины  
E-mail: yaviz@ukr.net

**Резюме:** С целью построения оптимальных по быстродействию и оборудованию арифметических устройств вычислительной техники, проведен анализ методов решения нелинейных уравнений в арифметических модулях вычислительных устройств. Предложен способ и алгоритм аппаратной реализации вычисления частного и обратной величины в сигнальных процессорах с увеличенным быстродействием и минимальными аппаратными затратами.

**Ключевые слова:** цифровая обработка сигналов, сигнальный процессор, архитектура, численные методы, производительность.

### 1. ВВЕДЕНИЕ

В настоящее время новые возможности, предоставленные разработчикам со стороны элементной базы, разрешили значительно повысить производительность средств цифровой обработки сигналов (ЦОС) и родиться другому, альтернативному подходу к построению высокопроизводительных вычислительных систем. Основная принципиальная особенность этого подхода состоит в построении таких средств обработки информации, которые вследствие некоторой избыточности, заложенной в их архитектуру, допускают возможность программно реконструироваться от одного алгоритма ЦОС к другому. Такой подход позволяет разрабатывать отдельные устройства сразу для целого класса задач ЦОС. Желание иметь более универсальные средства привело к созданию аналого-цифровых, программно управляющих процессоров, которые в дальнейшем начали называть сигнальными процессорами (СП).

До настоящего времени в алгоритмах цифровой обработки сигналов, которые были реализованы в сигнальных процессорах, основными операциями были умножение, сложение и вычитание. Вычисления нелинейной функции в операциях деления, извлечения квадратного корня, получения обратной величины, встречались крайне редко, не более 5% общего числа операций. Поэтому эти

операции выполнялись программными методами, что существенно не снижало быстродействия СП. С появлением новых, более сложных алгоритмов [1], в которых эти операции являются основными, возникла потребность их реализации на аппаратном уровне, поскольку программные методы стали тормозом на пути повышения производительности СП. В этих алгоритмах требуется анализировать энергетический спектр сигнала, вычислять модули комплексных чисел, ограничивать какой-либо параметр сигнала (амплитуду, частоту) путем деления на константу [2] и др. Иногда эти операции встречаются одновременно, например, алгоритм фазоразностной модуляции, что еще в большей степени влияет на производительность СП. Анализируя вышесказанное, приходим к выводу, что весьма актуальным становится вопрос поиска новых методов вычислений частного, обратной величины и способов их реализации.

### 2. ПОСТАНОВКА ЗАДАЧИ

В цифровой фильтрации методы вычисления сверток имеют ярко выраженный циклический характер. Это означает, что они выполняются не один раз, а многократно в цикле. Например, для получения выходного отсчета цифрового фильтра необходимо многократно перемножать входные отсчеты на коэффициенты, а результат суммировать в накапливающем сумматоре. И,

несмотря на то, что для большего числа операндов (отчетов сигнала) найден эффективный инструмент вычислений – алгоритм быстрого преобразования Фурье, который значительно сокращает число таких циклов, именно эти многократно выполняемые сложения, вычитания и умножения определяют быстродействие сигнального процессора. Таким образом, чтобы не усложнять архитектуру и не уменьшать быстродействие СП, алгоритм деления, следует реализовывать при помощи этих же операций на основе быстрых методов. Среди многочисленных методов ускорения операции деления, особый интерес вызывают синхронные методы ускорения, выполняемые “обходным” путем с использованием операции умножения. Поэтому, для решения поставленной задачи, необходимо проанализировать известные синхронные методы, позволяющие вычислять частное с использованием операций умножения, сложения и вычитания.

### 3. АНАЛИЗ СИНХРОННЫХ МЕТОДОВ

Известны методы деления, предложенные Стефанелли [3], основу которых составляют несколько алгоритмов получения обратной величины  $\frac{1}{C}$  двоичного делителя  $C$ , расположенного в диапазоне между 1/2 и 1. Эти методы могут быть расширены и для вычисления частного от деления двух чисел.

Формирование частного

$$Q \approx \frac{1}{C}$$

производится в два этапа – вначале частное вырабатывается в виде двоичного числа с избыточным представлением цифр, затем оно преобразовывается в простую двоичную форму.

В избыточном двоичном числе

$$X = x_1 \cdot 2^{-1} + x_2 \cdot 2^{-2} + \dots$$

в качестве цифр  $x_i$  используются, вообще говоря, любые положительные или отрицательные целые числа.

Основная идея метода состоит в том, что избыточные цифры подбираются таким образом, чтобы сумма членов, стоящих в одной колонке, в точности равнялась единице, т.е. очередному разряду произведения  $C \cdot Q$ . В работе [3] описано матричное устройство, которое весьма громоздкое, что вызвано, в первую очередь,

большим диапазоном возможных значений цифр.

Следующий рассматриваемый метод называют “гарвардской итерацией”. Этот метод можно применять только в том случае, если арифметический модуль СП обрабатывает нормализованные операнды с фиксированной запятой, причем текущие множители используют знаковый разряд как информационный.

Запишем формулу для вычисления частного:

$$C = \frac{N}{D}, \quad N < D, \quad D \neq 0, \quad (1)$$

где  $N$  – делимое, а  $D$  – делитель. Поскольку  $D$  – нормализованное число, то оно может изменяться в диапазоне  $\left(\frac{1}{2} \div 1\right)$ . Отсюда число

$D$  можно записать в виде:

$$D = 1 - x, \quad (2)$$

где  $x$  – дополнение к 1 для числа  $D$ . Подставив значение  $D$  в формулу (1), получим:

$$C = \frac{N}{1 - x}. \quad (3)$$

В формуле (3) числитель и знаменатель умножим на  $(1 + x)$ , тогда

$$C = \frac{N(1 + x)}{1 - x^2}. \quad (4)$$

Так как, согласно формулы (2), число  $x$  изменяется в диапазоне  $\left(0 \div \frac{1}{2}\right)$ , то  $x^2$  будет

изменяться в диапазоне  $\left(0 \div \frac{1}{4}\right)$ . Частное  $C$ , из

формулы (3), умножим  $n$  раз на сопряженные множители знаменателя:

$$C = \frac{N \cdot (1 + x) \cdot (1 + x^2) \cdot \dots \cdot (1 + x^{2^{n-1}})}{1 - x^{2^n}} \quad (5)$$

Если  $x$  изменяется в диапазоне  $\left(0 \div \frac{1}{2}\right)$ , то

$x^{2^n}$  будет изменяться в диапазоне  $\left(0 \div \frac{1}{2^{2^n}}\right)$ .

Очевидно, что  $\lim_{n \rightarrow \infty} \frac{1}{2^{2^n}} = 0$ . Значит знаменатель в формуле (5), при  $n \rightarrow \infty$ , стремится к 1.

Отсюда выражение (5) можно записать в виде

$$C \approx N \cdot (1+x) \cdot (1+x^2) \cdot \dots \cdot (1+x^{2^{n-1}}). \quad (6)$$

Таким образом, трудоемкое деление заменено на умножение, увеличенных на единицу, дополнений. Для увеличения быстродействия арифметического модуля СП, значение  $(1+x^{2^{n-1}})$  можно получать из таблиц. Этот метод целесообразно использовать в арифметических блоках СП, где разрядность операнда не превышает 8 бит, в остальных же случаях он проигрывает в быстродействии и оборудовании методу Ньютона.

Далее, для вычисления частного в СП, рассмотрим возможность применения быстрых итерационных методов. В работе [4] итерационный метод Ньютона был выбран для операции извлечения квадратного корня, что позволило значительно увеличить производительность арифметического модуля СП. Исходя из вышесказанного, оценим возможность применения итерационного метода Ньютона для вычисления частного.

Из работы [5] известно, что метод Ньютона или метод линеаризации определяется формулой:

$$X_{n+1} = X_n - \frac{f(x_n)}{f'(x)}, \quad (7)$$

$$f'(x_n) \neq 0, \quad n = 0, 1, 2, \dots$$

Пусть частное:

$$C = \frac{b}{d} = b \cdot \frac{1}{d}, \quad d \neq 0, \quad (8)$$

где  $d$  – делитель;  $b$  – делимое. Для вычисления обратной величины делителя  $\frac{1}{d} = y$  или для решения уравнения

$$f(y) = y^{-1} - d, \quad (9)$$

используем формулу (7):

$$y_{n+1} = y_n - \frac{y_n^{-1} - d}{(-1) \cdot y_n^{-2}} = y_n \cdot (2 - d \cdot y_n), \quad (10)$$

где  $y_n$  – начальное приближение;  $y_{n+1}$  – приближенное значение обратной величины делителя  $d$ . Подставляя значение  $y_{n+1}$  в формулу (8) получим приближенное значение частного  $C$ :

$$C \approx b \cdot y_{n+1} \approx b \cdot y_n \cdot (2 - d \cdot y_n) \quad (11)$$

Известно [5], что для погрешности  $Z = y - y^*$ , где  $y^*$  – приближенный корень уравнения (10), имеет место соотношение:

$$\lim_{n \rightarrow \infty} \frac{Z^{n+1}}{Z^{n^2}} = \frac{f''(y^*)}{2f'(y^*)} \quad (12)$$

Фактически соотношение (12) означает, что на каждой итерации погрешность возводится в квадрат, т.е. число верных знаков корня удваивается. Если  $\frac{f''(y^*)}{2f'(y^*)} \approx 1$ , то легко

показать, что при  $|Z| < 0,5$ , после пяти – шести итераций погрешность станет величиной порядка  $2^{-64}$ . Это наименьшее возможное значение погрешности при вычислениях на современных ЭВМ даже с удвоенной точностью. Заметим, что для получения столь малой погрешности в методе дихотомии потребовалось бы более 50 итераций [5].

Метод Ньютона сходится с квадратичной скоростью, что значительно выше, чем в других методах (метод простой итерации сходится со скоростью геометрической прогрессии). Кроме того, он имеет высокий показатель точности вычислений, при относительно малом количестве итераций. Поэтому метод Ньютона наиболее подходит в качестве базового для вычисления частного и обратной величины в арифметическом блоке сигнального процессора.

#### 4. ОСОБЕННОСТИ ПРИМЕНЕНИЯ

В большинстве случаев в арифметических блоках СП для операндов используется формат с фиксированной запятой, кроме того, на любой алгоритм, который реализуется в арифметическом блоке СП, накладываются жесткие временные ограничения, что имеет свои особенности при реализации алгоритма вычисления частного.

Для вычисления частного  $C$  по формуле (8), в формате с фиксированной запятой, необходимо, чтобы выполнялись условия  $d > b$  и  $d \neq 0$ . Как и при вычислении квадратного корня в [4], вычисление обратной величины  $\frac{1}{d}$  требует нормализации  $d$ . Тогда формулу (8) запишем в виде:

$$C = \frac{b}{d} = \frac{b \cdot H_0}{d \cdot H_0} = b \cdot H_0 \cdot \frac{1}{d'}, \quad (13)$$

где  $d' = d \cdot H_0$  – нормализованный делитель, который изменяется в диапазоне  $0,5 < d' \leq 1$ . Теперь обратную величину  $y_{n+1}$  можно записать следующим образом:

$$y_{n+1} = y_n \cdot (2 - d' \cdot y_n) \quad (14)$$

Код нормализации  $H_0$  и начальное условие  $K_0 = y_0 \approx \frac{1}{d'}$  являются представителями множеств кодов, которые использовались в алгоритме вычисления квадратного корня [4], что позволит не увеличивать аппаратные затраты при разработке арифметического модуля реализующего операции извлечения квадратного корня и деления. С учетом вышесказанного, запишем формулы для первой итерации при вычислении обратной величины:

$$y_1 = y_0 \cdot (2 - d' \cdot y_0) = K_0 \cdot (2 - d' \cdot K_0). \quad (15)$$

Значение частного будет вычислено по формуле (16) только после окончания итерационного процесса, т.е. после вычисления значения  $y_{n+1}$ :

$$C = b \cdot H_0 \cdot y_{n+1} \quad (16)$$

Количество итераций, которое нужно выполнить для получения необходимой точности результата, очевидно, зависит от выбора первого приближения. Поэтому для выбора начальных условий в арифметическом устройстве сигнального процессора необходимо использовать блоки постоянно запоминающих устройств (ПЗУ), которые смогут обеспечить требуемую точность  $y_n$ . Момент окончания операции извлечения корня определяется не путем сравнения разности  $|y_{n+1} - y_n|$  с некоторым допуском, а путем подсчета количества итераций, которые заранее известны в зависимости от разрядности используемых операндов. При выборе ПЗУ для хранения начальных условий  $y_0$ , необходимо стремиться к минимальной разрядности между числом разрядов операндов и числом адресных входов ПЗУ. Например, если операнды  $m$ -разрядные, то, для вычисления  $y_{n+1}$  только с одной

итерацией необходимо ПЗУ разрядностью  $\geq \frac{m}{2}$ ,

так как каждая итерация удваивает число верных разрядов.

Поскольку при каждом делении по любому из рассмотренных здесь методов нужно выполнять несколько умножений, то эти методы особенно эффективны в арифметических устройствах с быстрыми (например, матричными) умножителями, которые, как отмечалось ранее, являются обязательными в архитектуре СП.

С учетом вышесказанного и рекомендаций [6], можно составить обобщенный алгоритм рис.1 для местного устройства управления сигнального процессора. Данный алгоритм разработан с учетом ограничений для делителя и делимого, которые накладываются в формуле (13).

Допустим, что делитель  $d$  имеет  $m$  разрядов, причем количество разрядов, которые не подключены к адресным входам ПЗУ, обозначим переменной  $k$  с необходимым условием  $(m - k) > \frac{m}{2}$ . Тогда к адресным входам ПЗУ

будет подключено  $m - k$  разрядов делителя  $d$  или двоичный код  $d_0$ , при этом, на выходе ПЗУ получим  $m - k$  разрядное начальное условие  $K_0$ . В случае, если старшие разряды делителя  $d$  равняются нулю, т.е.  $d_0 = 0$ , то необходимо произвести коррекцию делителя  $d$  и делимого  $b$ , сдвинув их по разрядной сетке влево на  $2^{k+1}$  разрядов. В процессе выполнения итераций, к входам ПЗУ вместо двоичного кода  $d_0$  будут подключаться вычисленные значения  $y_n$  до тех пор, пока число итераций не станет равным, ранее заданному числу. Далее по алгоритму, используя код нормализации  $H_0$ , получаем нормализованный делитель  $d'$ , после чего, согласно формулам (14) и (15) выполняются итерационные шаги для вычисления приближенной обратной величины  $y_{n+1}$ . После окончания итерационного процесса, согласно формуле (16), выполняются заключительные операции для вычисления частного.

Так как современные СП обычно используют двухканальные арифметические модули при обработке комплексных операндов, далее рассмотрим возможность применения вышеописанного метода для вычисления комплексного частного.

Обычно операция деления двух комплексных чисел записывается следующим образом:

$$\dot{C} = \frac{\dot{A}}{\dot{B}} = \frac{\text{Re } A \cdot \text{Re } B + \text{Im } A \cdot \text{Im } B}{(\text{Re } B)^2 + (\text{Im } B)^2} + j \frac{\text{Re } B \cdot \text{Im } A - \text{Im } B \cdot \text{Re } A}{(\text{Re } B)^2 + (\text{Im } B)^2} \quad (17)$$

где  $\text{Re } A, \text{Re } B$  – действительные части, а  $\text{Im } A, \text{Im } B$

– мнимые части комплексных чисел  $\dot{A}$  и  $\dot{B}$  соответственно.

Кроме выполнения операции деления по формуле (17), существует способ деления, который можно построить на следующем предположении.

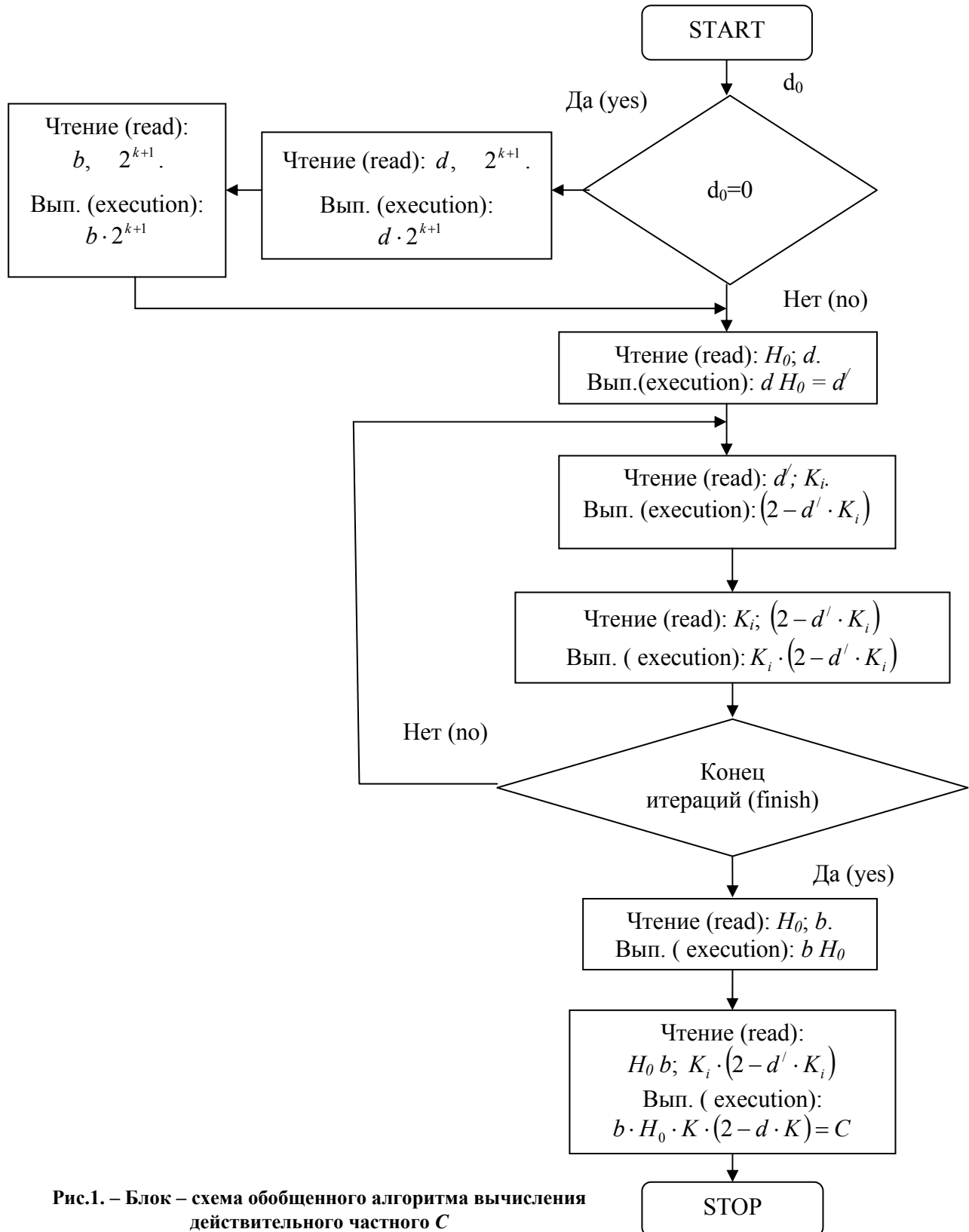


Рис.1. – Блок – схема обобщенного алгоритма вычисления действительного частного C

Пусть  $\dot{x}$  – частное,  $\dot{A}$  – делимое,  $\dot{B}$  – делитель. Тогда можно записать следующее равенство:

$$\dot{A} = \dot{x} \cdot \dot{B}. \quad (18)$$

Переходя от комплексного равенства (18) к двум вещественным равенствам, получим:

$$\begin{cases} \operatorname{Re} B \cdot \operatorname{Re} x - \operatorname{Im} B \cdot \operatorname{Im} x = \operatorname{Re} A \\ \operatorname{Im} B \cdot \operatorname{Re} x + \operatorname{Re} B \cdot \operatorname{Im} x = \operatorname{Im} A \end{cases} \quad (19)$$

где  $\operatorname{Re} A$ ,  $\operatorname{Re} x$ ,  $\operatorname{Re} B$  – действительные части, а  $\operatorname{Im} A$ ,  $\operatorname{Im} x$ ,  $\operatorname{Im} B$  – мнимые части комплексных чисел  $\dot{A}$ ,  $\dot{x}$  и  $\dot{B}$  соответственно. Система уравнений (19) представляет собой систему линейных алгебраических уравнений второго порядка, которая в матричной форме имеет вид

$$\begin{pmatrix} \operatorname{Re} B & -\operatorname{Im} B \\ \operatorname{Im} B & \operatorname{Re} B \end{pmatrix} \times \begin{pmatrix} \operatorname{Re} x \\ \operatorname{Im} x \end{pmatrix} = \begin{pmatrix} \operatorname{Re} A \\ \operatorname{Im} A \end{pmatrix}. \quad (20)$$

Таким образом, операция деления может рассматриваться как решение системы линейных алгебраических уравнений. Нетрудно убедиться, что решение указанной системы при помощи определителей нецелесообразно, так как это приведет к вычислениям аналогичным (17), причем решения подобных систем выполняются методом “цифра за цифрой”. По сравнению с ускоренной операцией умножения такой метод деления будет отрицательно влиять на производительность СП, работающего в реальном времени. Поэтому целесообразно реализовать операцию комплексного деления, используя операцию умножения и численные методы. Для реализации этой цели, обратимся к формуле (17), но запишем ее в виде двух вещественных уравнений:

$$\begin{cases} \operatorname{Re} C = \frac{\operatorname{Re} A \cdot \operatorname{Re} B + \operatorname{Im} A \cdot \operatorname{Im} B}{(\operatorname{Re} B)^2 + (\operatorname{Im} B)^2} = \frac{R}{D} \\ \operatorname{Im} C = \frac{\operatorname{Re} B \cdot \operatorname{Im} A - \operatorname{Im} B \cdot \operatorname{Re} A}{(\operatorname{Re} B)^2 + (\operatorname{Im} B)^2} = \frac{I}{D} \end{cases}, \quad (21)$$

где  $R$  и  $I$  – действительные числители в  $\operatorname{Re} C$  и  $\operatorname{Im} C$  соответственно;  $D$  – действительный знаменатель. На первом этапе, в двухканальном арифметическом модуле, имеющего два

умножителя – накопителя, определяем значения  $R$ ,  $I$  и  $D$ . Затем, в соответствии с предложенным алгоритмом (рис.1), вычисляем значения  $\operatorname{Re} C$  и  $\operatorname{Im} C$  на соответственном канале.

## 5. ВЫВОДЫ

Применение предложенных методов в сигнальных процессорах позволяет трудоемкое деление заменить на умножение, с минимальными затратами оборудования, что даст возможность:

- разработать арифметический модуль с минимальными аппаратными затратами;
- не увеличивать командный цикл процессора;
- получить временной выигрыш по сравнению с программной реализацией;
- вычислять частное комплексного числа в двухканальных арифметических блоках;
- выполнять вычисления основного алгоритма в реальном времени.

Кроме этого, разработанный арифметический модуль, реализующий предложенный алгоритм вычисления частного, может быть использован как отдельный вычислительный элемент core – генератора, в системе автоматического проектирования Xilinx и др.

Дальнейшие исследования в этой области направлены на создание новых архитектур сигнальных процессоров, которые позволят расширить круг задач ЦОС, вычисляемых аппаратно, с высоким быстродействием.

## 6. СПИСОК ЛИТЕРАТУРЫ

- [1] Gold B., Rader C.M. *Digital Processing of Signals, Ch. 4*. McGraw-Hill, 1969.
- [2] А.В. Палагин, М.В. Семотюк, Е.Н. Чичирин, К.П. Сосненко. Acoustic Commander – интегрированная операционная среда для измерения и расчета акустических параметров. *Компьютерные средства, сети и системы*, 2009, №8, с. 3-10.
- [3] Stefanelly R. A suggestion for a high-speed parallel binary divider. *IEEE Trans. Comput.*, 1972, v. C-21, No. 1, pp. 42-55.
- [4] Я.Е. Визор. Ускоренная реализация решения нелинейной функции в сигнальных процессорах и микроконтроллерах. *Проблемы информатизации и управления*, 2009, №3, с. 20-25.
- [5] Н.С. Бахвалов. *Численные методы*. – М.: Наука, 1975.
- [6] Ахо Д., Хопкрафт Д. *Построение и анализ вычислительных алгоритмов*. – М.: Мир, 1979.



**Палагин Александр Васильевич**, академик Национальной академии наук Украины, доктор технических наук, профессор, заместитель директора Института кибернетики В.М. Глушкова, Национальной академии наук Украины

**Семотюк Мирослав Васильевич**, кандидат технических наук, ведущий научный сотрудник Института кибернетики НАН Украины. В 1973 г. окончил Азейбарджанский политехнический институт по специальности вычислительная техника. Автор 52 опубликованных научных работ, имеет 2 патента на изобретения.

Область научных интересов: теория универсальных и специализированных компьютерных систем; алгебра и теория чисел; теория цифровой обработки сигналов.



**Визор Ярослав Евстафьевич**, кандидат технических наук, старший научный сотрудник Института кибернетики НАН Украины. В 1979 г. окончил электромеханический факультет Львовского политехнического института.

Автор 28 опубликованных научных работ, имеет 11 патентов на изобретения.

Область научных интересов: теория универсальных и специализированных компьютерных систем; алгебра и теория чисел; теория цифровой обработки сигналов.



**Чичирин Евгений Николаевич**, кандидат технических наук, старший научный сотрудник Института кибернетики имени В.М. Глушкова НАН Украины. В 1974 г. окончил Киевский политехнический институт по специальности вычислительная техника. Автор 20

опубликованных научных работ, имеет 7 патентов на изобретения.

Область научных интересов: сжатие данных, теория кодирования, спектральный анализ, цифровые сигнальные процессоры и системы.



## SOME ASPECTS OF INCREASING THE PROCESSING POWER OF THE DIGITAL SIGNAL PROCESSORS

Alexander Palagin, Miroslav Semotiuk, Yaroslav Vizor, Eugeniy Chichirin

V.M.Glushkov Institute of Cybernetics  
of the National Academy of Sciences of Ukraine  
E-mail: yaviz@ukr.net

**Abstract:** *An analysis of the non-linear equation solving methods used by the arithmetic units of the computing devices was conducted in order to facilitate creation of the arithmetic units which would be optimal in terms of processing power and hardware overhead. A method and algorithm of a hardware implementation of computing the particular and the reciprocal in the digital signal processors at higher speed and minimal hardware overhead.*

**Keywords:** *digital signal processing, signal processor, architecture, calculus of approximations, computing speed.*

### 1. INTRODUCTION

Computing non-linear functions in the division operations, square roots and the reciprocal operations used to be quite scarce until recently, amounting to less than 5% of the total number of operations. Thus, they were easily carried out in software mode, having low negative impact of the signal processor speed. With the advent of the new, more complex algorithms [1] employing these operations as the basic ones, a need to ensure a hardware implementation of the said operations arose, for the software methods became the brake hindering the drive to increase the signal processors' (SP) speed. Such algorithms deal with analyzing the signal spread spectrum, computing the complex number modulus or limiting one of the signal's parameters (like amplitude or frequency, for instance) by dividing it by a constant [2], etc. Having analyzed the stated above, the authors came to a conclusion that the search for the new methods of computing the particular and the reciprocal and ways of their implementation has become necessary.

### 2. THE TASK AT HAND

Analysis of the contemporary digital signal processing (DSP) algorithms shows that the DSP's processing speed is most influenced by the addition, subtraction and multiplication operations, which are frequently used. Thus, in order not to make the DSP architecture overly complex and avoid decreasing the DSP processing speed, the division algorithm should be implemented using the same operations to

carry the computing out in accordance with advanced methods. Among the numerous methods of speeding up the division operation execution the most interesting ones are the synchronous methods which use the "roundabout" way employing the multiplication operation. Thus, in order to solve the problem at hand, the known synchronous methods of computing the particular with the aid of the multiplication, addition and subtraction operations are to be analyzed.

### 3. ANALYSIS OF THE SYNCHRONOUS METHODS

The division methods proposed by Stefanelli [3] are the first to be reviewed in the article. The main idea of the method boils down to arranging the redundant digits so that the sum of the members belonging to one column would be equal to the current position of the  $C \cdot Q$  multiplication product. The research paper [3] describes a matrix device which is quite cumbersome due to the wide range of acceptable digital values.

The following method is called "Harvard iterative method" and is used when the DSP arithmetic unit is working on the fixed point normalized operands while the current multipliers are using the sign bit as an information one. As a result of replacements the formula (1) is transformed into expression (6), where  $x$  is a complement to 1 for  $D$  (2) divisor,  $N$  – dividend, and  $C$  – quotient. This approach is feasible for the DSP arithmetic units supporting 8-bit operands or lower, in all the other



cases it turns out to be less effective than the Newton method.

The Newton iterative method application is also reviewed as a way of computing the quotients in a DSP. This method is employed in the paper [4] for implementing the square root operation, leading to a noticeable increase in the DSP arithmetic unit's processing speed. By using the expression (7) defining the Newton iterative method, the authors have arrived at the iteration formula (11) for computing the real quotient  $C$ . It has been proven that the Newton method converges at quadratic speed which is much faster than the other methods mentioned above. Besides, it also has a quite high computing accuracy at a comparatively low number of iterations. That's why the Newton method's been chosen as the base for computing the particular and the reciprocal by the DSP arithmetic unit.

#### 4. APPLICATION FEATURES

The main thing to keep in mind in order to ensure correct computation of a particular using the formula (8) is the usage of normalization code  $H_0$  and the initial reciprocal value  $K_0$ , stored in read-only memory (ROM). These quotients belong to the set of codes used in the square root computation algorithm [4], thus allowing for lower hardware overhead when designing a arithmetic unit performing the square root and division operations. After the necessary transformations of formula (8), the authors have arrived at expression (15) for the first iteration of the  $y_l$  reciprocal value computation. The particular value is computed using the formula (16) only after the iteration process is completed, i.e. when the value  $y_{n+1}$  is obtained.

The number of iterations needed to obtain the required result accuracy is known beforehand and depends on the choice of first approximation. When storing the initial conditions  $y_0$  in the ROM a note should be taken to keep the difference between the operand bit width and the number of ROM address outputs minimal. Following the recommendations listed in paper [6], a generalized algorithm (Fig.1) for computing the real particular has been created.

The possibility of employing the Newton method for computing the complex particular in the double-channel DSP arithmetic units is described below. The authors have also described a way of using the real particular computation algorithm for implementing the complex number division operations by transforming the complex equation (17) into a system of two real equations (21) in the paper. At the initial stage, the double-channel DSP arithmetic unit equipped with two multiplication –

storage units computes the R, I and D values. Then, in accordance to the algorithm (Fig.1), the corresponding channel computes the ReC and ImC.

#### 5. CONCLUSION

Employing the proposed method in the DSPs allows to replace the resource-taxing division process with multiplication, having minimal hardware overhead which, in turn, allows to:

- develop an arithmetic unit with minimal hardware overhead;
- refrain from prolonging the processor's instruction cycle;
- obtain a better execution time in comparison to software implementation;
- perform the particular computation with the double-channel DSP arithmetic units;
- perform real-time execution of the core algorithm;

Besides that, the developed arithmetic unit employing the proposed particular computation algorithm can be used as a separate computing core element – generator in the Xilinx CAD system, etc.

Further research in this field shall be directed towards developing the new signal processor architectures capable of widening the spectrum of digital signal processing tasks solved by high-performance hardware-based systems.

#### 6. REFERENCES

- [1] Gold B., Rader C.M., *Didital Processing of Signals, Ch. 4*. McGraw–Hill, 1969.
- [2] A.V. Palagin, M.V. Semotiuk, E.N. Chichirin, K.P. Sosnenko. Acoustic Commander – an integrated operation environment for measurement and calculation of acoustic properties. *Computers, networks and systems*, 2009, No. №8, pp. 3-10. (in Russian).
- [3] Stefanelly R. A suggestion for a high-speed parallel binary divider. *IEEE Trans. Comput.*, 1972, v. C-21, No. 1, pp. 42-55.
- [4] Y.E. Vizor. Accelerated implementation of non-linear function computing in signal processors and microcontrollers. *Informatisation and management problems*, 2009, No. 3, pp. 20-25. (in Russian).
- [5] N.S. Bakhvalov. *Numeric methods*. Moscow: Nauka, 1975. (in Russian).
- [6] A.V. Aho, J.E. Hopcraft. *The Design and Analysis of Computer Algorithms*. Addison-Wesley, Reading, Mass., 1974.