

# Самосинхронное устройство умножения-сложения гигафлопсного класса: методологические аспекты

И.А. Соколов, Ю.А. Степченков, Ю.В. Рождественский, Ю.Г. Дьяченко

Учреждение Российской академии наук Институт проблем информатики РАН (ИПИ РАН),

[\[ISokolov, YStepchenkov, YRogdest, YDiachenko\]@ipiran.ru](mailto:[ISokolov, YStepchenkov, YRogdest, YDiachenko]@ipiran.ru)

**Аннотация** — Представлены подходы к проектированию самосинхронной (СС) аппаратуры и рассмотрены условия внутрисистемной интеграции синхронных и СС-устройств в рамках супер-ЭВМ на примере разработки СС-устройства умножения-сложения гигафлопсного класса, соответствующего стандарту IEEE 754 и выполняющего либо одну операцию двойной точности, либо одновременно две операции одинарной точности над входными операндами. Устройство реализовано по КМОП-технологии с проектными нормами 65 нм. В зависимости от исполнения оно работает с асинхронным или синхронным окружением и обеспечивает среднюю производительность на уровне не менее 1 гигафлопс при времени задержки формирования результата относительно поступления входных операндов не более 6 нс.

**Ключевые слова** — самосинхронная схема, супер-ЭВМ, умножитель с накоплением, сумматор, конвейер.

## 1. ВВЕДЕНИЕ

Современные супер-ЭВМ петафлопсного класса используют менее 35% вычислительной мощности для непосредственных вычислений [1]. Остальные ресурсы расходуются на обеспечение достоверности результатов вычислений. При этом их наработка на отказ составляет ~55 часов (данные на 2013 год соответствуют супер-ЭВМ, содержащим до 100000 ядер вычислительных устройств). Для супер-ЭВМ эксафлопсного класса, в которых число ядер повышается почти на три порядка, а нормы проектирования СБИС сокращаются в 2-3 раза, показатели надёжности вычислений и наработка на отказ могут оказаться существенно хуже. Решение этой проблемы возможно либо кардинальным изменением технологического базиса супер-ЭВМ, либо использованием аппаратных методов контроля надёжности и саморемонта вычислительных средств в темпе реальных вычислений. Создание и глубокая технологическая проработка нового базиса вычислительных систем (ВС) видится в весьма отдалённой перспективе. В то же время для второго варианта решения можно уже сейчас предложить СС-схемотехнику, достаточно хорошо проработанную и отвечающую выдвигаемым требованиям.

Эта работа посвящена расширению класса СС-блоков вычислительного ядра для супер-ЭВМ. Ранее был разработан и изготовлен квазисамосинхронный блок деления/извлечения квадратного корня ([2], [3]) 64/32-разрядной арифметики в технологии 180 и 65

нм. Здесь излагаются основные принципы проектирования СС-устройства умножения-сложения 64/32-разрядности гигафлопсного класса для супер-ЭВМ, учитывающие особенности работы СС-схем.

Несмотря на теоретически доказанные и практически подтвержденные преимущества СС-изделий (в том числе разработанных, изготовленных и испытанных авторами настоящего доклада), коммерческих СС-изделий немного [4], а сведения об их успешной эксплуатации в рамках крупных ВС отсутствуют. Представляется, что одной из причин является проблема встраивания СС-изделий в глобальное синхронное окружение. В ряде работ В.И. Варшавского была предложена методология построения крупных ВС в виде GALA-систем (Global Asynchronous / Local Arbitrary). Если ее принять на вооружение конструкторам супер-ЭВМ, это решило бы проблему внедрения СС-изделий. На уровне системного интерфейса супер-ЭВМ реализуется асинхронный интерфейс обмена данными взаимодействующих устройств, а на уровне локального интерфейса – произвольный. При этом на уровне локального интерфейса проектировщик может использовать широкую гамму решений – от реализации полной самосинхронизации до использования локального генератора. Все зависит от цели проектируемого изделия.

Но именно при реализации такой стратегической линии построения супер-ЭВМ обеспечивается возможность постепенного внедрения СС-изделий (по мере появления их на рынке) в общую структуру для достижения требуемых параметров (надёжности и/или энергоэффективности).

Операция "умножение двух операндов и сложение произведения с третьим операндом" с девяностых годов прошлого века стала широко внедряться в микропроцессоры общего и специального назначения. В виде одной инструкции она реализована в микропроцессорах IBM POWER1 (1990 г.) и старше, Fujitsu SPARC64 (1995 г.), HP PA-8000 (1996 г.), Intel Itanium (2001 г.), IBM Cell (2005 г.), в процессорах AMD (2012 г.), в графических процессорах NVIDIA Tesla T10 (GPGPU) [5]. В 2008 году она вошла в стандарт IEEE 754 [6].

Эта операция обычно выполняется быстрее, чем пара последовательных инструкций умножения и сложения, и точнее за счет отсутствия промежуточного округления произведения.

В настоящее время известно множество публикаций, посвященных разработке умножителей с такого типа накоплением, известных в литературе как Fused Multiply-Add (FMA). Несмотря на то, что операция FMA хорошо изучена и реализована во множестве устройств, она остается предметом пристального внимания разработчиков, учитывающих вклад устройства FMA в общую производительность центрального Вычислителя, аппаратные затраты на его реализацию и потребляемую мощность. В основном рассматриваются синхронные устройства FMA [7], [8], но в последние годы все больше публикаций посвящается асинхронным FMA, например, [9], [10].

Анализ источников [9], [10] показал, что они опираются на использование так называемой спекулятивной индикации и слабых транзисторов. Использование спекулятивной индикации (окончание переходных процессов индицируется только в некотором критическом пути) позволяет резко сократить затраты оборудования на индикаторную часть, но приводит к потере самопроверяемости относительно константных неисправностей. Использование слабых транзисторов позволяет снизить общее число транзисторов, но приводит к существенному снижению уровня помехоустойчивости аппаратуры.

Более того, использование указанных приемов уменьшает зону работоспособности СС-изделий, особенно в области предельно низких питающих напряжений. Это нивелирует преимущество СС-схем, позволяющих существенно снизить динамическую и статическую составляющие энергопотребления для той части аппаратуры супер-ЭВМ, которая в настоящий момент либо не используется вообще (не задействована в текущей задаче), либо может быть использована с существенно меньшей производительностью, путем динамического снижения питающего напряжения [11].

Указанные виды СС-схем неприемлемы для использования в составе супер-ЭВМ. В рамках сложившейся на сегодня классификации целям супер-ЭВМ отвечают только два класса СС-схем:

- схемы, правильная работа которых не зависит от задержек элементов и от задержек в проводах при условии, что разница задержек проводов после разветвления меньше минимально возможной задержки базового элемента библиотеки; такие схемы мы называем независимыми от задержек элементов (НЗЭ); в западной классификации им соответствуют схемы *speed-independent (SI)* или *quasi delay-insensitive*;

- схемы, правильная работа которых не зависит ни от задержек элементов, ни от задержек в соединяющих проводах; по сложившейся западной классификации такие схемы называются *delay-insensitive (DI)*.

Ярким представителем DI-схем являются NCL-схемы (Null Convention Logic [12]). Проведенный авторами сравнительный схемотехнический анализ НЗЭ- и NCL-схем [13] показал, что "НЗЭ схемы, разрабатываемые в соответствии с методологией, продвигаемой ИПИ РАН, обладают меньшими аппаратными затра-

тами (в 4,49 раза при реализации двоичного счетчика, в 1,13 раза при реализации умножителя 4x4, до 2 раз при реализации более простых логических схем), большей производительностью и меньшим энергопотреблением по сравнению с NCL-схемами". Поэтому именно их целесообразно использовать в качестве схемотехнического базиса супер-ЭВМ эксафлопсного класса.

НЗЭ-схемотехника использует запрос-ответную дисциплину взаимодействия блоков, СС-кодирование информационных сигналов и развитую систему индикации хода вычислительного процесса. Поэтому появление любой константной неисправности в любой цепи такой схемы приводит к остановке вычислений и формированию соответствующего индикатора, локализуя проблему. Это позволяет практически в темпе вычислений использовать резервный аппаратный ресурс и продолжить безошибочную работу всего устройства (осуществляется саморемонт). В ряде случаев, применение НЗЭ-схемотехники обеспечит увеличение в несколько раз реальной производительности ВС в составе супер-ЭВМ эксафлопсного класса.

Платой за такие преимущества является аппаратная избыточность и дополнительные временные затраты на индикацию и дополнительную фазу "гашения" в работе самосинхронных элементов. Грамотное проектирование НЗЭ-схем существенно снижает эту избыточность, а в ряде случаев [14] приводит к получению даже лучших результатов, чем в синхронных схемах.

Цель данного доклада – оценка подходов к разработке SI-устройства умножения-сложения (SIFMA) по стандарту IEEE 754 [6], обладающего расширенными функциональными возможностями и ориентированного на использование в супер-ЭВМ.

## II. ОСОБЕННОСТИ РАЗРАБАТЫВАЕМОГО SIFMA

Область применения данного SIFMA выдвигает в качестве основного требования минимальное энергопотребление при достаточно высокой производительности [1]. Предполагаемая производительность 1 гигафлопс. Она определяется сравнительно невысокой тактовой частотой для машин эксафлопсного класса и большим числом узлов SIFMA на одну СБИС. Анализ минимаксной кривой, построенной в координатах "энергопотребление – площадь" в соответствии с методикой, приведенной в [15], применительно к технологии 65-нм и к производительности SIFMA, позволил определить основные характеристики прототипа, выбрать структурную схему SIFMA, представленную на рис. 1, и определить число ступеней в конвейерной реализации на 1 гигафлопс, равной 8. Предлагаемый вариант SIFMA разрабатывался в рамках стандартной технологии 65 нм с ограничением латентности операций в 6 нс.

SIFMA предназначен для потоковой обработки входных операндов. Операнды двойной точности представляются как 64-разрядные числа, а одинарные – как 32-разрядные числа. Это позволяет использовать 64-разрядные операнды для передачи сразу двух 32-

разрядных чисел. Тем самым создается предпосылка для опциональной обработки одним устройством не только операндов двойной точности, но и одновременной обработки двух однотипных операций одинарной точности.

Разработанный SIFMA выполняет операции в соответствии со стандартом IEEE 754 со следующими уточнениями:

- на вход поступают нормализованные операнды,
- перемножаемые операнды не являются нулевыми,
- тройка операндов содержит либо три числа двойной точности, либо шесть чисел одинарной точности; в последнем случае одновременно выполняются две операции над тройками операндов одинарной точности.

Кроме того, SIFMA опционально выполняет сразу две операции: сложение произведения двух операндов с третьим операндом и/или вычитание произведения из третьего операнда.

Рассматриваемое устройство SIFMA реализует следующую совокупность действий по обработке входных операндов [7]:

- 1) вычисление экспоненты результата;
- 2) умножение мантисс первых двух операндов;
- 3) выравнивание мантиссы третьего операнда;
- 4) инвертирование выровненной мантиссы третьего операнда в случае фактического вычитания произведения из третьего операнда или наоборот;
- 5) сложение и/или вычитание мантис произведения и третьего операнда;
- 6) получение модуля результата при отрицательном значении суммы и/или разности;
- 7) нормализация модуля суммы и/или разности;
- 8) округление нормализованного результата;
- 9) постнормализация.

Главной особенностью описываемого SIFMA является его принадлежность к классу устройств, не зависящих от задержек элементов [2], [3], правильно работающих при любых задержках элементов, на которых они реализованы. Время выполнения операции в нем определяется не тактовым сигналом, а условиями эксплуатации (напряжением питания, температурой окружающей среды) и характером обрабатываемых операндов.

Достигаются такие характеристики с помощью кодирования сигналов специальными СС-кодами и средств индикации завершения отдельных этапов обработки данных. В подавляющем большинстве НЗЭ-схем используется парафазный код и индикаторы на Г-триггерах или С-элементах.

Фундаментальной проблемой проектирования СС-схем для супер-ЭВМ является необходимость индикации завершения каждого этапа вычислений для всех разрядов машинного слова перед началом последующего этапа обработки. Такие индикаторы требуют су-

щественных аппаратных затрат и привносят дополнительную задержку в работу процессора ЭВМ вследствие большой длины машинного слова и избыточности СС-кодирования. Очевидно, что решение этой проблемы возможно двумя путями:

- сокращением числа индицируемых элементов;
- снижением избыточности СС-кодирования.

В качестве средства решения этих проблем при разработке SIFMA исследовалась целесообразность использования позиционной целочисленной симметричной троичной системы счисления, которую характеризуют следующие свойства [16]:

- отсутствие проблемы округления (для отдельных режимов округления достаточно просто отбросить ненужные цифры);
- укорочение таблицы умножения;
- упрощение проблемы переноса при суммировании.

Итогом исследования стало использование троичной арифметики [-1, 0, +1] для представления суммы двух парафазных сигналов при построении сумматоров в дереве Уоллеса блока умножения мантисс. Для представления сигналов в троичной арифметике был разработан специальный СС-код. Это позволило снизить количество ступеней дерева Уоллеса с 7 до 4 для множителя разрядности 53\*53, реализующего алгоритм Бута в формате Radix2. В результате сократились аппаратные затраты на индикацию и снизилось время выполнения операции умножения.

### III. ПРОБЛЕМЫ ИНДИКАЦИИ SIFMA

Корректная работа любой SI-схемы обеспечивается индицированием окончания всех переключений схемы. Схемы, в которых индицируются выходы всех без исключения элементов в обеих фазах работы (спейсерной и рабочей), называются полумодулярными [17]. Однако индикаторная подсхема, как правило, существенно замедляет работу полумодулярной многоуровневой схемы. Критичным с точки зрения быстродействия многоуровневых самосинхронных устройств является реализация общего индикатора устройства, объединяющего все поразрядные индикаторные сигналы в один общий индикаторный сигнал.

Особенностью предлагаемого подхода к реализации SIFMA являются следующие принципы организации индикации:

- необходимая и достаточная, но упрощенная индикация рабочей фазы каждой ступени конвейера;
- использование спекулятивной индикации для ускорения взаимодействия ступеней конвейера.

С учетом того, что результат обработки данных комбинационной схемой (КС) в каждой ступени конвейера сохраняется в выходном регистре ступени и там индицируется в обеих фазах работы, нет необходимости индицировать КС в рабочей фазе. Используемое

парафазное и специальное троичное СС-кодирование данных на всех этапах обработки гарантирует однократное переключение выходов КС из спейсера в рабочее состояние при переключении всей ступени конвейера из спейсера в рабочую фазу. Поэтому появление рабочего состояния после спейсера на информационных входах всех разрядов выходного регистра ступени гарантирует готовность результата.

Напротив, при переключении КС в спейсер индицируются все элементы, так как перед переходом в следующее рабочее состояние нужно быть уверенным в том, что все элементы схемы переключились в спейсер. В противном случае возможны "гонки" и неоднократные срабатывания выходов элементов, что является нарушением принципов самосинхронной реализации.

Поскольку КС в каждой ступени конвейера SIFMA - сложное функциональное многоуровневое устройство, такая упрощенная индикация существенно сокращает аппаратные затраты и ускоряет формирование индикаторных сигналов, участвующих в запрос-ответном взаимодействии ступеней конвейера.

Спекулятивный индикатор отслеживает окончание переключения только элементов, стоящих на критическом пути обработки данных ступенью конвейера. Фактически это индикатор одного разряда тракта обработки многоуровневых данных. В многоуровневой схеме он формируется с гораздо меньшей задержкой, чем общий индикатор. Поскольку взаимодействие ступеней конвейера SI-схемы основано на двухфазной дисциплине работы [17], спекулятивный индикатор текущей ступени может использоваться в качестве сигнала разрешения переключения в противоположную фазу работы предшествующей ступени. Это позволяет предыдущей ступени начать переключение в очередную фазу, не дожидаясь подтверждения окончания переключения всех разрядов текущей ступени, что существенно (на 30-50%, в зависимости от сложности схемы и разрядности шин данных) ускоряет работу конвейера.

Логическим обоснованием такого подхода к индикации многоуровневых SI-схем служит следующее. Спекулятивный индикатор контролирует один из аналоговых разрядов тракта обработки данных, в том числе - разряд выходного регистра. При правильном выборе соответствующего разряда в подавляющем большинстве случаев он будет переключаться не раньше (а если и раньше, то не намного) остальных разрядов схемы. Поэтому спекулятивный индикатор как бы отражает работу всех разрядов ступени. Общий индикатор оказывается более задержанным из-за того, что он "собирает" индикаторные сигналы со всех разрядов схемы с помощью "дерева" индикаторных элементов. Чем больше разрядов в схеме, тем более задержанным окажется общий индикаторный сигнал в сравнении со спекулятивным индикатором.

Стопроцентная самопроверяемость конвейера SIFMA обеспечивается использованием полного индикаторного выхода каждой ступени в качестве дополни-

тельного разрешения переключения этой ступени конвейера в противоположную фазу работы. Такая схема по-прежнему индицирует константные неисправности в полном объеме, хотя и не обеспечивает бесперебойной работы при *любой* задержках элементов. С практической точки зрения, она обладает всеми преимуществами самосинхронных схем перед синхронными аналогами. С точки зрения полноты индикации SIFMA не является полумодулярным. Но при правильном выборе источника спекулятивного индикатора будет работоспособным в широком диапазоне условий эксплуатации (напряжения питания, температуры), технологических и электрических параметров компонентов и обязательно остановится при возникновении константной неисправности.

#### IV. СТРУКТУРНАЯ СХЕМА SIFMA

Как основа для реализации супер-ЭВМ блок SIFMA должен обладать достаточным быстродействием. В синхронных схемах проблема быстрого сложного вычислительного тракта наиболее эффективно решается с помощью конвейеризации вычислений. При этом разрядность схемы, если она достаточно регулярная, не играет роли.

В НЗЭ-схемах конвейеризация также используется. Но здесь количество разрядов в схеме напрямую влияет на ее быстродействие, так как приводит к усложнению и замедлению подсхемы индикации. Кроме того, конвейеризация умножителя требует использования регистров для хранения промежуточных результатов, разрядность которых намного превышает разрядность входных операндов. Это дополнительно усложняет как саму схему, так и индикаторную подсхему, и обеспечивает лишь небольшое увеличение быстродействия. Другими словами, повышение быстродействия самой сложной части SIFMA - умножителя с помощью его конвейеризации оказывается "нерентабельным".

Поэтому было решено для достижения максимального быстродействия SIFMA использовать два блока умножителя, работающих параллельно, как показано на структурной схеме (рис. 1).

Здесь входные операнды - 64-разрядные обрабатываемые числа X, Y, Z и 6-разрядные признаки операции R (тип округления, точность и характер операции). Тип округления - двухразрядный признак, определяющий способ округления результата в соответствии с требованиями стандарта IEEE 754. Характер операции - двухразрядный признак, задающий количество и тип операций, выполняемых над операндами. Использование признака характера операции позволяет минимизировать энергопотребление SIFMA в случаях, когда требуется выполнение только одной операции (сложения или вычитания) над произведением и третьим операндом.

Следует отметить, что параллельно вычисляются только произведения операндов X и Y из двух последовательных троек операндов, поступающих на вход SIFMA. Дальнейшие вычисления выполняются поочередно для каждой тройки входных операндов в одном

общем тракте обработки данных. Такое решение обеспечивает наилучшее соотношение "быстродействие / аппаратные затраты" при реализации SIFMA.

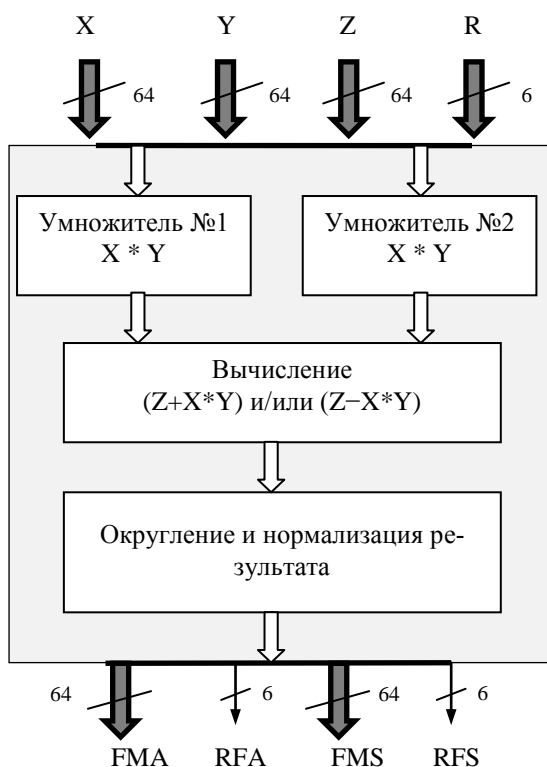


Рис. 1. Структурная схема SIFMA

Время выполнения операции блоком SIFMA, как и любым НЗЭ-устройством, зависит от многих факторов: условий эксплуатации (напряжения питания, температуры окружающей среды), типа операции, вида операндов. Функциональные блоки SIFMA работают под управлением сигналов запрос-ответного взаимодействия и не требуют сигналов синхронизации.

Однако при работе с синхронным окружением возникает необходимость приема входных операндов во фронту сигнала системной частоты. В этом случае отсутствие потерь на приеме обеспечивается двумя способами:

- подбором системной частоты (исходя из наилучшего случая), при которой SIFMA заведомо успеет обработать любую тройку операндов при любых предусмотренных условиях эксплуатации;
- использованием устройств сопряжения синхронных и асинхронных устройств, предотвращающих потерю операндов за счет приостановки "накачки" данных со стороны синхронного окружения.

Второй способ - более эффективный и быстродействующий, поскольку в его рамках легко реализовать буферизацию данных на входе и выходе SIFMA.

Для обеспечения максимального быстродействия SIFMA при работе с синхронным окружением схему на рис. 1 предлагается дополнить входным и выход-

ным НЗЭ FIFO, как показано на рис. 2. Самосинхронное FIFO накапливает поступающие тройки операндов и признаки операции, обеспечивая формирование предупреждения о заполнении FIFO. Сигнал предупреждения Stop устанавливается (Stop=1) в том случае, если занята входная ячейка FIFO. Это происходит при приеме каждой тройки операндов, как показано на рис. 3; здесь Clock – сигнал глобальной синхронизации, Clk – тактовый вход устройства SIFMA. С продвижением операндов к выходу из FIFO по мере обработки предыдущих операндов входная ячейка освобождается, и сигнал Stop снимается (Stop=0).

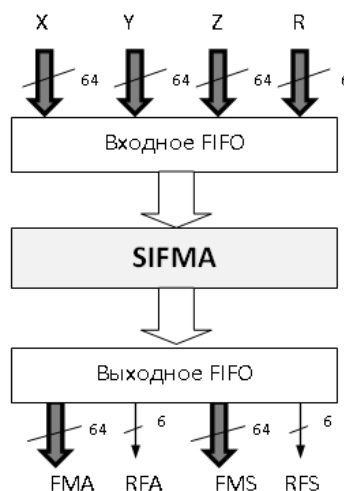


Рис. 2. Структурная схема SIFMA с обрамлением

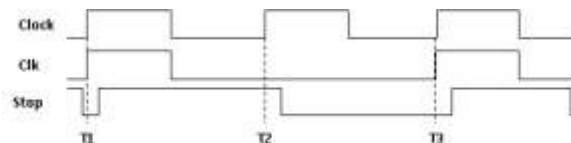


Рис. 3. Проблема входного интерфейса

Это дает синхронному окружению возможность вовремя приостановить подачу данных на вход SIFMA. На рис. 3 в моменты времени T1 и T3 SIFMA оказывается готовым к приему очередных данных, и формируется тактовый сигнал Clk. А в момент времени T2 снятие сигнала Stop опаздывает по отношению к активному фронту тактового сигнала Clock и заставляет синхронное окружение приостановить запись данных в SIFMA.

Следующие блоки SIFMA последовательно обрабатывают поступающие данные в самосинхронном режиме: по мере готовности данные передаются из текущего блока в следующий, а текущий блок переходит в режим ожидания данных из предыдущего блока.

FIFO на выходе SIFMA также предотвращает потерю производительности SIFMA, когда время формирования очередного результата превысит период системной частоты, из-за чего один из циклов чтения резуль-

тата синхронным окружением может быть пропущен, как показано на рис. 4.

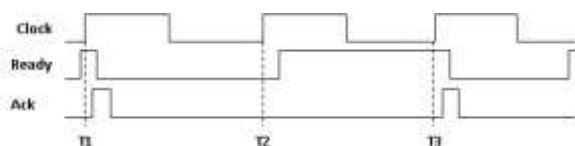


Рис 4. Проблема выходного интерфейса

Сигнал Ready индицирует готовность результата на выходе SIFMA. Синхронное окружение фиксирует его по активному фронту тактового сигнала Clock и сигналом Ack=1 подтверждает прием данных. В моменты времени T1 и T3 результат оказывается готовым перед появлением очередного активного фронта Clock, а в момент времени T2 опаздывает по отношению к активному фронту тактового сигнала. Из-за этого возникает "холостой" такт: результат выполнения очередной операции с выхода SIFMA считывается с опозданием на один такт.

Использование входного и выходного НЗЭ FIFO позволяет решить проблемы максимально быстрой "накачки" SIFMA входными операндами и в полной мере использовать способность SIFMA выполнять операцию умножения-сложения за разное время в зависимости от типа операции, вида операндов и условий эксплуатации.

В соответствии с описанными принципами были разработаны два варианта SIFMA [18] – для синхронного и асинхронного окружения. Они подтвердили требуемый уровень производительности и энергопотребления.

## V. ЗАКЛЮЧЕНИЕ

В супер-ЭВМ эксафлопсного класса, когда число ядер достигает сотен миллионов, практически необходимо существенно усиливать аппаратную составляющую средств обеспечения надёжности и достоверности результатов вычислений. Самосинхронная схемотехника, предложенная для реализации основных вычислительных узлов, обеспечивает высокую эффективность в решении этой задачи.

Впервые в отечественной и зарубежной практике предпринята попытка разработки действительно 64/32-разрядного SIFMA-устройства в виде схемы, поведение которой не зависит от задержек элементов и в проходах до точки разветвления.

## ПОДДЕРЖКА

Исследование выполнено при финансовой поддержке РФФИ в рамках научных проектов №№ 13-07-12062 офи\_м и 13-07-12068 офи\_м, а также при частичной финансовой поддержке Программы фундаментальных исследований ОНИТ РАН за 2013 г. (проект 1.5).

## ЛИТЕРАТУРА

[1] URL: <http://book.itep.ru/10/supercomp.htm> (дата обращения: 15.01.2014).

- [2] Степченко Ю.А., Дьяченко Ю.Г., Бобков С.Г. Квасисамосинхронный вычислитель: методологические и алгоритмические аспекты // Проблемы разработки перспективных микро- и нанoeлектронных систем – 2008. Сб. трудов / под общ. ред. академика РАН А.Л. Стемповского. М.: ИППМ РАН, 2008. С. 441-446.
- [3] Дьяченко Ю.Г., Рождественский Ю.В., Морозов Н.В., Степченко Д.Ю. Квасисамосинхронный вычислитель: практическая реализация // Проблемы разработки перспективных микро- и нанoeлектронных систем – 2008. Сб. трудов / под общ. ред. академика РАН А.Л. Стемповского. М.: ИППМ РАН, 2008. С. 435-440.
- [4] Bink A. and York R. ARM996HS: The First Licensable, Clockless 32-Bit Processor Core // IEEE Micro. 2007. V. 27. № 2. P. 58-68.
- [5] URL: <http://www.seobuilding.ru/wiki/FMA> (дата обращения: 12.01.2014).
- [6] IEEE Computer Society. IEEE Standard for Floating-Point Arithmetic IEEE Std. 754-2008. doi:10.1109/IEEE STD. 2008.4610935.
- [7] Pillai R.V.K., Shah S.Y.A., Al-Khalili A.J., Al-Khalili D. Low Power Floating Point MAFs – A Comparative Study // Sixth International Symposium on Signal Processing and its Applications. 2001. V. 1. P. 284-287.
- [8] Seidel P.-M. Multiple Path IEEE Floating-Point Fused Multiply-Add // Proceedings of the 46th IEEE International Midwest Symposium on Circuits and Systems. 2003. P. 1359-1362.
- [9] Noche J.R., Araneta J.C. An asynchronous IEEE floating-point arithmetic unit // Proceedings of Science Diliman. 2007. V. 19. № 2. P. 12-22.
- [10] Manohar R., Sheikh B.R. Operand-Optimized Asynchronous Floating-Point Units and Method of Use Therefor. US patent, № 20130124592. May 2013.
- [11] Руткевич А., Бумагин А., Гондарь А., Штенко Г. и др. Методы снижения энергопотребления в строго самосинхронных микропроцессорных схемах // Компоненты и технологии. 2009. № 9. С. 109-114.
- [12] Smith S.C. and Jia Di. Designing Asynchronous Circuits using NULL Convention Logic (NCL) // Synthesis Lectures on Digital Circuits and Systems. 2009. V. 4. № 1. P. 61-73.
- [13] Соколов И.А., Степченко Ю.А., Бобков С.Г., Захаров В.Н., Дьяченко Ю.Г., Рождественский Ю.В., Сурков А.В. Базис реализации супер-ЭВМ эксафлопсного класса // Информатика и ее применения. 2014. Т. 8. № 1. 28 с.
- [14] Степченко Ю.А., Петрухин В.С., Дьяченко Ю.Г. Опыт разработки самосинхронного ядра микроконтроллера на базовом матричном кристалле // Проблемы разработки перспективных микроэлектронных систем – 2005. Сборник трудов / под общ. ред. академика РАН А.Л. Стемповского. М.: ИППМ РАН, 2005. С. 235-242.
- [15] Galal S. and Horowitz M. Energy-Efficient Floating-Point Unit Design // IEEE Transactions on computers. 2011. V. 60. № 7. P. 913-922.
- [16] Троичная система счисления – Википедия. URL:[http://en.wikipedia.org/wiki/Ternary\\_numerical\\_system8F](http://en.wikipedia.org/wiki/Ternary_numerical_system8F) (дата обр. 14.01.2014).
- [17] Автоматное управление асинхронными процессами в ЭВМ и дискретных системах / под ред. В.И. Варшавского. М.: Наука, 1986. 400 с.
- [18] Степченко Ю.А., Рождественский Ю.В., Дьяченко Ю.Г., Морозов Н.В., Степченко Д.Ю. Самосинхронное устройство умножения-сложения гигафлопсного класса: варианты реализации // В настоящем сборнике трудов.