Толочанов С.В.

ОРГАНИЗАЦИЯ ВЫЧИСЛЕНИЙ НЕ ПО ФОН-НЕЙМАНУ

ВВЕДЕНИЕ

Предметом данной статьи являются некие в определенной степени универсальные технические решения и методы, связанные с реализацией вычислительных устройств, построенных не на принципах фон-Неймана.

1 ИСХОДНЫЕ ПРЕДПОСЫЛКИ

Для понимания истоков предлагаемых методов расскажу некую свою собственную предысторию. Мне кажется, что возврат к истокам может быть иногда весьма полезным для того, чтобы понять, а вдруг был еще другой путь развития, отличающийся от выбранного и развиваемого далее.

Заранее прошу прощения за, может быть, излишние и не совсем технические подробности.

Для того, кто хочет сразу перейти к конкретным аспектам предлагаемых технических решений и методов, этот раздел можно не читать.

На заре своей инженерной деятельности в области вычислительной техники (начало 70-х годов), когда микропроцессоры только-только робко о себе заявляли, вычислительные устройства разрабатывали (по крайней мере, в нашей организации) на базе конечных автоматов Мили или Мура. В качестве элементной базы применялись микросхемы малой степени интеграции, в основном это были триггеры различных типов и комбинационная логика.

Сам процесс разработки велся в соответствии с классическими принципами фон-Неймана.

Для меня принципы фон-Неймана сводятся к двум постулатам:

- аппаратное разделение вычислительного устройства на управляемую (операционную) и управляющую части;
- последовательное во времени выполнение мелких операций, приводящее к решению задачи в соответствии с программой, реализующей исходный требуемый алгоритм.

Сначала определялась структура управляемой (операционной) части, в которой выделялись точки (узлы схемы), требующие подачи на них сигналов от управляющей части. Далее в соответствии с требуемым алгоритмом или, по-другому, временной диаграммой синтезировалась управляющая часть.

Таким методом было разработано более десятка различных вычислительных устройств, а сам процесс разработки был существенным образом формализован, что, как правило, всегда приводит к ускорению сроков разработки и уменьшению количества ошибок.

Надо отметить тот существенный факт, что практически главным критерием разработки, помимо, естественно, корректного функционирования устройства, было минимальное количество оборудования (микросхем и других радиоэлектронных компонентов), необходимого для реализации требуемой задачи.

Так все это катилось своим чередом, пока не появился у нас на работе молодой специалист, только что окончивший ВУЗ. Правда специальность у него была такова, что не читали им в то время ни принципов фон-Неймана, ни программирования и других дисциплин классической (для тех времен) вычислительной техники. У него была специальность «Радиотехника».

Так получилось, причем во многом благодаря его собственной инициативе, а также моего желания проверить его, так сказать, в деле, что я ему поручил разработать уже готовое к тому времени устройство, разработанное вышеописанными методами.

Каково же было мое удивление, когда через некоторое время он разработал нормально работающее устройство, которое содержало примерно на 15-20 % меньше оборудования, чем ранее разработанное по классическим принципам. Этот молодой специалист разработал требуемое устройство просто, что называется «в лоб» по требуемой временной диаграмме выходных и входных сигналов.

Этот факт поначалу меня в некоторой степени озадачил. В результате определенных размышлений я для себя нашел некое объяснение данному факту. Я подсчитал суммарное количество триггеров в управляющей и управляемой частях устройства, которое оказалось несколько большим, чем во вновь разработанном устройстве. Соответственно, и микросхем комбинационной логики было меньше.

Те элементы памяти (триггеры), которые входили в состав управляющей части дублировали функции триггеров, входящих в состав управляемой части устройства.

Вывод, который я для себя сделал на то время, был таков:

С точки зрения затрат оборудования разделение аппаратной части вычислительного устройства на управляющую и управляемую часть не оптимально.

Дальнейшее развитие микроэлектроники, появление микропроцессоров полностью снивелировало этот недостаток одного из принципов фон-Неймана.

Тем не менее, для меня этот факт остался в памяти.

В последнее время появилось достаточно большое количество публикаций о системном кризисе вычислительных архитектур, построенных на принципах фон-Неймана.

По-моему, основа этого кризиса лежит во втором принципе (см. выше).

Далее попытаюсь изложить свой подход.

2 НЕКОТОРЫЕ ОБОСНОВАНИЯ МОЕГО ПОДХОДА

Основным исходным тезисом для дальнейших рассуждений является следующее:

ПОЧЕМУ РАЗРАБОТЧИКИ, А ПОТОМ И ПРОИЗВОДИТЕЛИ СРЕДСТВ ЛЮБОЙ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ В БУКВАЛЬНОМ СМЫСЛЕ ЗАСТАВЛЯЮТ ПОЛЬЗОВАТЕЛЕЙ (ПОТРЕБИТЕЛЕЙ) ПРИМЕНЯТЬ ТЕ ЖЕ АППАРАТНО-ПРОГРАММНЫЕ СРЕДСТВА, КОТОРЫЕ ПРИМЕНЯЛИСЬ ПРИ ИХ РАЗРАБОТКЕ?

3 ИЗЛОЖЕНИЕ СУТИ МЕТОДИКИ

3.1 Общие замечания

Здесь мы не будем рассматривать методы получения конечных и пригодных для какой-либо реализации формул и зависимостей, т.е. некие предметные (прикладные) в этом смысле задачи.

Нас не будет интересовать, каким образом получены реализуемые зависимости.

Дальнейшее изложение будет идти только в смысле методов реализации этих известных соотношений.

Конечной целью предлагаемой ниже методики является существенное повышение быстродействия при некоем универсальном подходе к реализации вычислений.

Дальнейшее изложение приводится для реализации предлагаемой методикой алгоритмов (в общем смысле понятия алгоритма, как такового, здесь – нет) на примере цифровых фильтров **(подчеркиваю, это только пример, не более того)** в силу их хорошей математической проработанности и распространенности, т.е. прозрачности для понимания предлагаемой методики.

3.2 Методика

Например, известна формула для реализации линейного нерекурсивного цифрового фильтра (ЦФ) с конечной импульсной характеристикой (КИХ):

$$y(n) = \sum_{i=0}^{m} b_i \cdot x(n-i),$$

где x(n) и y(n) – отсчеты, соответственно, входного и выходного сигналов в моменты времени n;

b – коэффициенты ЦФ;

m – порядок данного ЦФ.

В целях упрощения понимания в качестве примера реализуем вычисление соотношения для нерекурсивного ЦФ 2-го порядка по предлагаемой методике, т.е.

$$y(n)=b1*x(n-1)+b0*x(n) \qquad (1)$$

Предположим, что написана, отлажена и протестирована программа для реализации соотношения (1).

Причем совершенно безразлично, на каких аппаратных и при использовании каких программных средств эта программа написана и функционирует. Это может быть персональный компьютер, система на базе контроллера или сигнального процессора или любая другая системная платформа.

Время выполнения этой программы совершенно не критично, главное, чтобы она корректно вычисляла соотношение (1) с необходимой точностью.

Далее необходимо написать программу, которая позволяла бы путем подачи на вход программы, реализующей вычисление соотношения (1), полного набора входных пар (в рассматриваемом случае) x(n), x(n-1), фиксацию (запоминание) результатов вычислений для каждого значения входной пары в некоторой области памяти.

Другими словами, необходимо получить некую полную таблицу результатов при всех возможных значениях пар (в рассматриваемом примере) входных аргументов, которые в данном случае являются адресами ячеек памяти, хранящих результаты вычислений соотношения (1).

Необходимо подчеркнуть, что время заполнения такой таблицы результатов также абсолютно не критично, поскольку процесс заполнения таблицы происходит на этапе разработки, а не, так сказать, в «боевом» или, другими словами, режиме применения.

Например, при разрядности входов x(n), x(n-1) в 8 бит объем такой таблицы составит 64Кслов (разрядность результата определяется необходимой точностью вычисления соотношения (1) и может быть равна в данном случае, например, 8, 16 бит или любой другой требуемой величине).

Далее каким-либо способом необходимо записать полученную таблицу в память устройства, которое будет реализовывать методику в «боевом» режиме. При этом время записи также не критично, если это не связано с необходимостью изменения содержимого в процессе «боевого» применении.

4 СПОСОБЫ РЕАЛИЗАЦИИ

На рис. 1 представлена схема, реализующая вышеизложенную методику для разрядности входов в 1 байт.

Здесь и в других случаях не показаны цепи записи таблицы результатов в память, т.к. в каждом конкретном применении, для каждого типа памяти, в качестве которой можно применять любую память (SRAM, DRAM, ROM, EPROM, Flash и др.), эти цепи записи будут отличаться.

Во всех приведенных здесь и ниже вариантах реализации сигнал «синхр.» обеспечивает синхронизацию записи входной информации в момент времени, когда входной информационный сигнал сформирован и достоверен.

Схемотехника «СхУпр» обеспечивает внутреннюю синхронизацию устройств схемы и зависит в первую очередь от типа применяемой памяти.

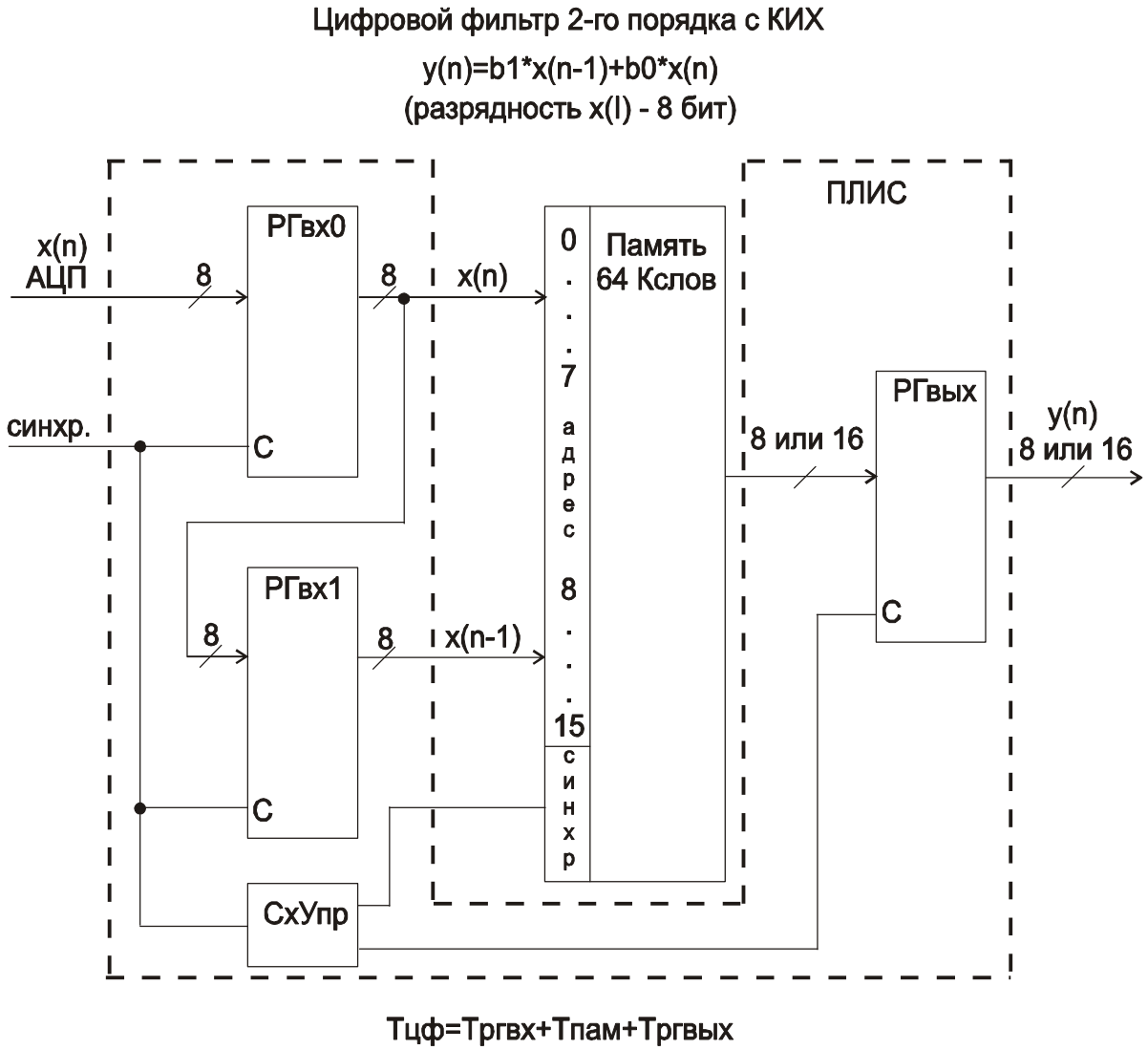

Рис. 1

При реализации устройств, «окружающих» память, при помощи микросхем ПЛИС можно достигнуть существенной экономии в оборудовании при высоком быстродействии.

Если объем памяти ПЛИС достаточен, то предлагаемые схемотехнические решения можно реализовать без применения «внешней» памяти.

Главной проблемой практической реализации данной методики является объем необходимой памяти результатов.

Например, при разрядности x(n), равной 16 (2 байта), объем памяти оставит 2^{32}=4 Гслова, что на первый взгляд ставит крест на реализации методики.

Такая же проблема возникает, например, при увеличении порядка ЦФ, реализуемых предлагаемым методом.

Однако, в обоих случаях имеется подход к решению этой проблемы при относительно небольших затратах дополнительного оборудования и увеличении времени выполнения Тцф на единицы нс.

На рис. 2 показана схема реализации ЦФ 4-го порядка, из которой видно, что проблема требуемого большого объема памяти решается введением в схему дополнительной памяти такого же объема и сумматора.

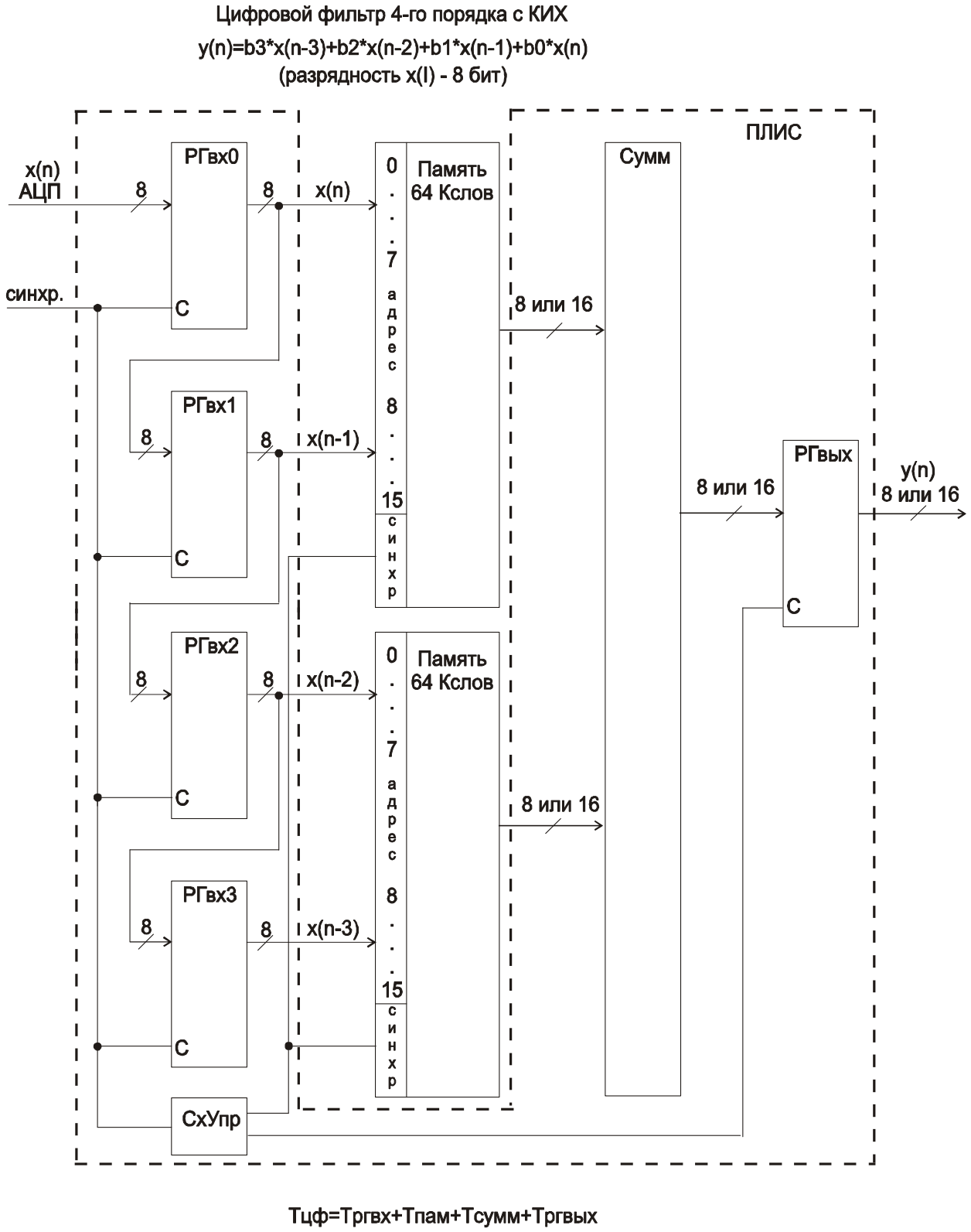

Рис. 2

На рис. 3 показана схема реализации ЦФ при разрядности аргументов 16 бит.

Аргумент представляется суммой двух байтовых аргументов с умножением старшего байта на $2^8 = 256$. Данное действие реализуется в ПЛИС путем программирования связей на входах сумматора с учетом «сдвига» на 8 бит (имеется ввиду «жесткий» сдвиг, т.е. не занимающий времени вычисления).

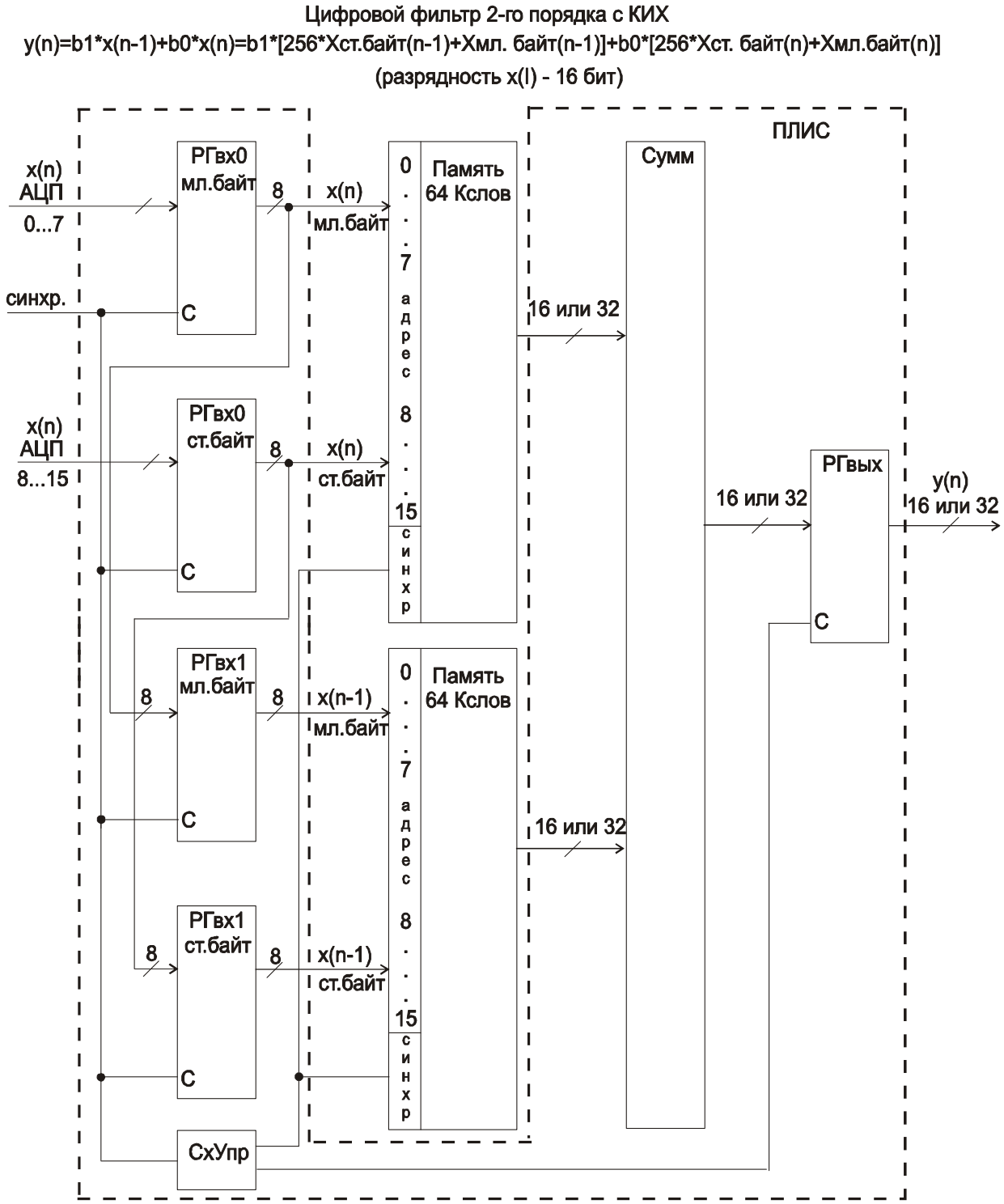

Рис. 3

На рис. 4 показана схема реализации адаптивного ЦФ, в которой коэффициенты b_i могут меняться извне (из управляющего процессора) в процессе адаптивной фильтрации.

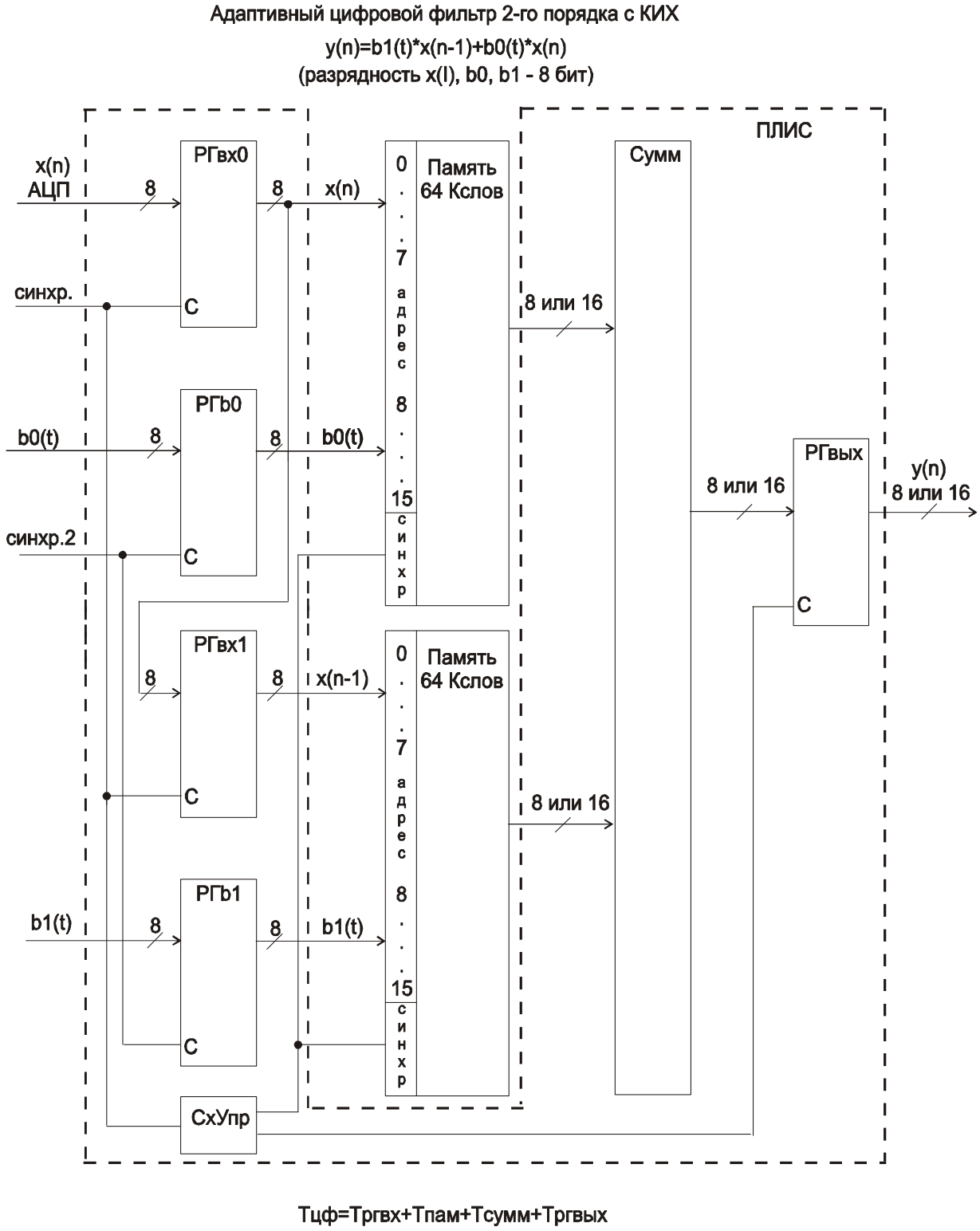

Рис. 4

На рис. 5 показана схема реализации рекурсивного ЦФ, очевидно, здесь дополнительных комментариев не требуется.

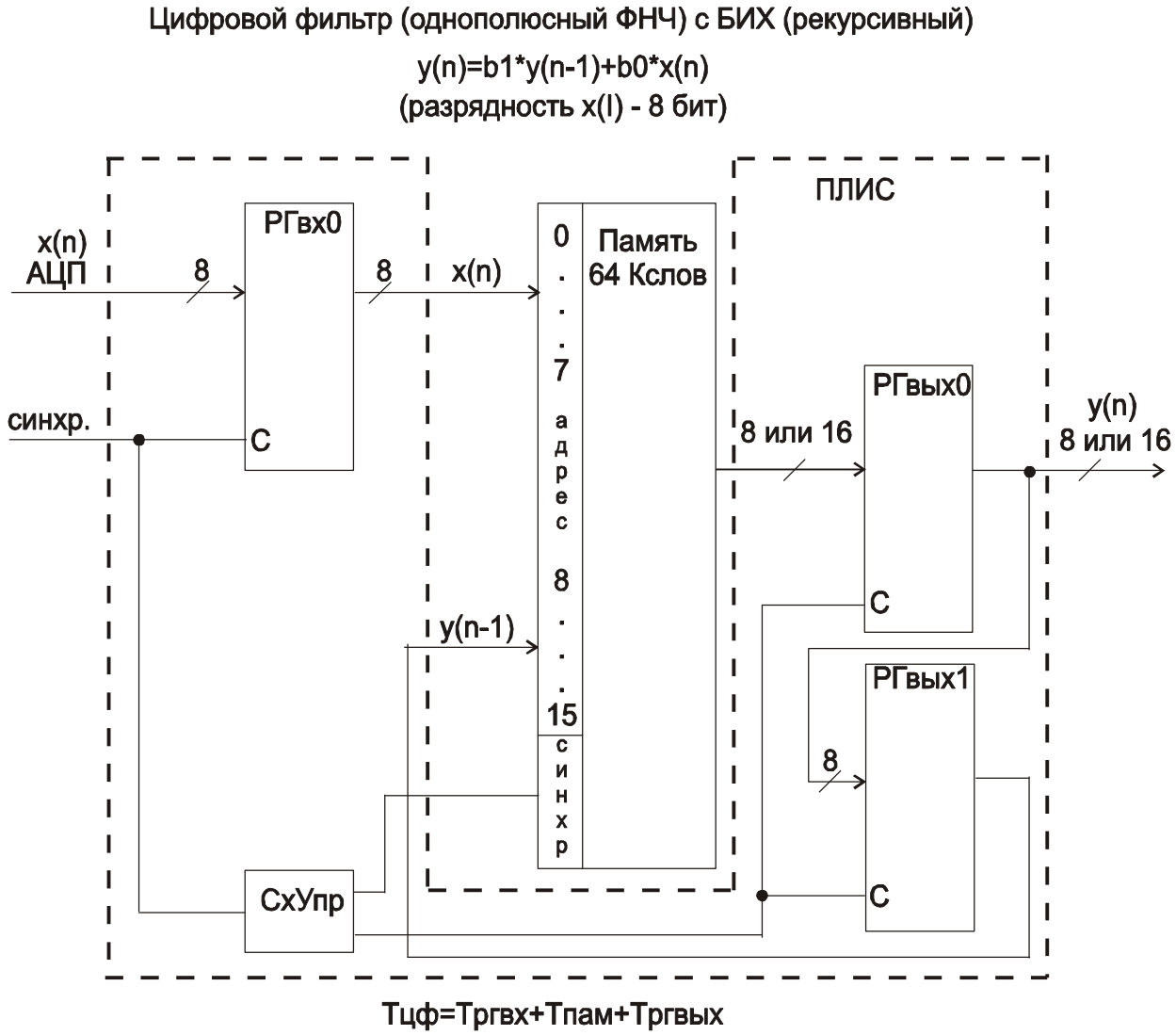

Рис. 5

5. ОЦЕНКА БЫСТРОДЕЙСТВИЯ

Для оценки быстродействия (времени выполнения ЦФ) необходимо определиться с быстродействием составных частей предлагаемой схемотехнической реализации.

Например, в случае применения в качестве памяти статических ОЗУ (SRAM) Тпам составляет 10…15 нс.

При применении ПЛИС значения Тргвх, Тргвых, Тсумм составляют единицы нс.

Даже реализация ЦФ на самых современных сигнальных процессорах не может дать сколько-нибудь приближающегося к предлагаемой методике быстродействия.

6. ОБЛАСТИ ПРИМЕНЕНИЯ

Предлагаемая методика может быть применена при решении задач цифровой обработки сигналов, обработки изображений, радиолокации, шифрования и др. подобных сферах.

При определенной работе этот метод может позволить, не применяя импортную дорогостоящую микроэлектронную технику, получать результаты, превышающие по быстродействию импортные аналоги, при этом не требовать жутких вложений в промышленность для достижения уровня «их» (имеется ввиду, в первую очередь, «американосовский» микронный уровень микроэлектроники) хваленых технологий, достигнутых, в первую очередь, за счет ограбления всего остального мира и определенной технической идеологии (имеется ввиду фон-неймановская структура построения вычислительных средств).

Если предлагаемый подход разрабатывать дальше, то, особенно, в области цифровой обработки сигналов, обработки изображений, радиолокации и др. подобных сферах, где играет роль точность вычисления, а не сложность выдаваемых управленческих сигналов, можно будет просто использовать «их» готовое ПО, и реализовывать необходимые задачи мною предлагаемыми средствами и методикой без применения «их» суперсложных микросхем, для собственного производства которых никаких нынешних наших (российских) денег не хватит.

7. НЕКОТОРЫЕ ДОПОЛНИТЕЛЬНЫЕ СООБРАЖЕНИЯ И ПЕРСПЕКТИВЫ

1). Из самого предлагаемого принципа построения такого рода вычислителей можно видеть, что при увеличении порядка ЦФ или разрядности обрабатываемых входных сигналов время выполнения Тцф не меняется, только увеличивается объем требуемой памяти и сложность схемотехники ПЛИС, которая может привести к увеличению их количества.

При этом совершенно не обязательно применять отдельную (не входящую в состав современных ПЛИС память, если таковой достаточно в самих ПЛИСах).

2). Дальнейшая перспектива развития такого подхода может заключаться в некоем «объединении» в одном таком вычислителе следующих за цифровой фильтрацией алгоритмов (обычно это преобразование Фурье) с собственно фильтрацией. Это может дать еще больший выигрыш в быстродействии при практически тех же затратах оборудования.

Возможно (особенно для преобразования Фурье) необходимо будет применять некое сочетание классического подхода, на котором может быть реализован управляющий процессор, с предлагаемым.

Под классической реализацией здесь и далее понимаются архитектуры типа фон-Неймана, гарвардская, смешанная (сигнальные процессоры фирмы Analog devices) и другие подобные.

3). При определенных условиях можно предлагаемые схемотехнические решения реализовать чисто программными средствами на классических системных платформах.

В данном случае целесообразность такой реализации определяется наличием выигрыша во времени при выполнении предлагаемым и классическим способом.

Кроме того, в этом случае существенно возрастают доступные ресурсы памяти при использовании, например, винчестера или других накопителей при условии наличия выигрыша во времени выполнения при их использовании.

4). Из самого предлагаемого способа можно увидеть, что структура ПЛИС, предложенных для применения, имеет в достаточно степени регулярный и во многом повторяющийся характер.

Сей факт в перспективе дает возможность разработки и применения специализированных БИС жесткой логики в различных возможных вариантах (например, отдельно регистры, отдельно сумматоры или какое-либо их сочетание с возможностями каскадирования), что еще более увеличит быстродействие, а также позволит увеличить степень их интеграции из-за отсутствия в них аппаратной избыточности, присущей универсальным ПЛИС.

При этом в качестве еще одной перспективы можно отметить возможность уменьшения зависимости от импортных микросхем, например, сигнальных процессоров и не только.

5). Также необходимо отметить определенную универсальность метода, выражающуюся в том, что для схожих в определенном смысле задач аппаратная структура не меняется, т.е. функции меняются только путем перезаписи таблицы результатов для иных реализуемых соотношений.

6). **У автора имеются предварительно проработанные подходы для реализации предлагаемыми методами функций многих переменных, а также нелинейных функций.**

Имеются определенные подходы к реализации алгоритмов, для которых практически невозможно получить какую-либо аналитическую форму представления реализуемых этими алгоритмами соотношений.

7). Предлагаемые подходы (по моему мнению) имеют возможность модернизировать существующую обработку информации с помощью традиционных нейросетевых технологий.

Имеются первоначальные предложения и на этот счет.

С уважением, и надеждой на любой отклик. Особенно, на компетентный.

Я желаю всем, кто прочитал то, что я написал выше, всех благ, здоровья и успехов во всех делах.