

Проектирование энергоэффективных модулярных КИХ фильтров на базе редуцированных мультиконстантных умножителей

Д.В. Тельпухов, А.С. Михмель, Р.А. Соловьев

Институт проблем проектирования в микроэлектронике РАН

Аннотация: В настоящее время особенную актуальность обретают вопросы разработки энергоэффективных аппаратных блоков для устройств цифровой обработки сигналов (ЦОС). Это связано с бурным расцветом носимой электроники (wearable electronics), интернета вещей (Internet of Things, IoT), сетевых и телекоммуникационных систем. Ключевым узлом многих устройств ЦОС является фильтры с конечной импульсной характеристикой (КИХ). Поэтому неудивительно, что в настоящее время большое число научных работ посвящено построению энергоэффективных КИХ фильтров. В статье предложен оригинальный подход к решению поставленной задачи. В качестве методологического базиса была выбрана модулярная арифметика, уже зарекомендовавшая себя как эффективный математический аппарат для построения быстродействующих устройств ЦОС. Другим решением было использование транспонированной формы КИХ фильтра и методов построения редуцированных блоков мультиконстантных умножителей. В экспериментальной части продемонстрирована эффективность методов редукции блоков мультиконстантного умножения с точки зрения энергопотребления фильтров. Также в статье сформулированы рекомендации по использованию предложенных методов для конкретных реализаций КИХ фильтров.

Ключевые слова: модулярный КИХ-фильтр, мультиконстантный умножитель, транспонированная форма, энергопотребление, выделенная мощность.

Введение

Фильтры с конечной импульсной характеристикой (КИХ-фильтры) являются основой большей части устройств цифровой обработки сигналов. Наибольшее применение они находят в системах беспроводной связи, Wi-Fi роутерах и т.д. Из-за увеличения тактовых частот и объемов передаваемого трафика, растёт потребляемая энергия таких устройств, поэтому производители ищут возможности для оптимизации энергопотребления. В данной статье приводятся результаты исследования энергоэффективности различных архитектур аппаратных КИХ-фильтров с фиксированными коэффициентами.

КИХ-фильтры

Фильтр с конечной импульсной характеристикой, по своей сути, является линейной сверткой входной последовательности некоторых цифровых отсчетов с последовательностью коэффициентов фильтра. Фильтры могут быть с фиксированными и изменяемыми коэффициентами. Задача выбора тех или иных коэффициентов фильтра - сложная, и в нашей работе не рассматривается. В настоящее время существует большое количество программных продуктов, которые позволяют рассчитывать коэффициенты фильтра для различных задач [1].

Абстрагируясь от значений коэффициентов, обратимся непосредственно к вычислению линейной свертки. Формула для ее вычисления выглядит следующим образом:

$$s(n) = a * b = \sum_{m=0}^n a(m) * b(n-m), n = 0 \dots N + M - 2$$

Архитектуры для вычисления линейных сверток могут быть совершенно различными. Выделяют несколько типов архитектур:

- Последовательная;
- Параллельная;
- Последовательно-параллельная.

Последовательная схема характеризуется малым числом вычислительных блоков, интенсивным обменом с памятью и низкой производительностью. В крайнем проявлении эта схема представляет собой умножитель с накоплением и управляющее устройство, которое обеспечивает загрузку нужных коэффициентов из памяти. В этом случае, для нахождения одного выходного отсчета требуется N тактов. Этот метод реализуется программным способом на сигнальных процессорах или компьютерах общего назначения.

В случае, если производительности DSP процессора не хватает, то фильтр реализуют аппаратно, используя параллельные архитектуры. Параллельные схемы эксплуатируют метод конвейеризации, разделяя этапы конвейера регистрами. В данной работе рассматриваются высокоскоростные параллельные КИХ-фильтры реализованные в аппаратном виде. Каноническая форма КИХ фильтра выглядит следующим образом (рис. 1):

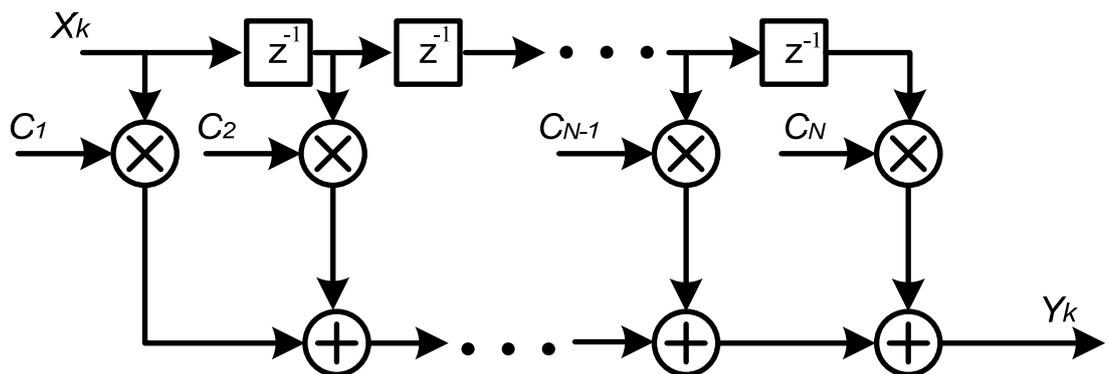


Рис. 1. – Каноническая форма параллельного КИХ-фильтра

Этот же фильтр можно реализовать в транспонированной форме (рис. 2).

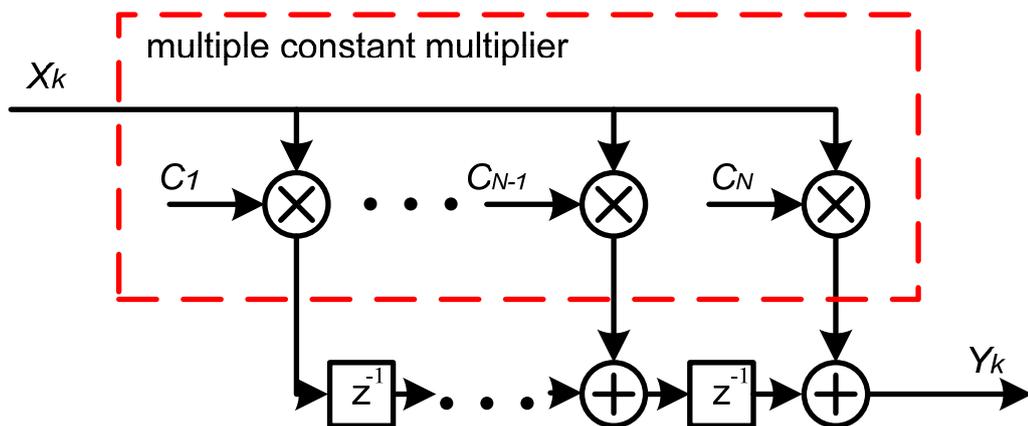


Рис. 2. – Транспонированная форма КИХ-фильтра

Главным преимуществом данной формы является наличие мультиконстантного множителя, который можно оптимизировать с целью уменьшения площади и/или энергопотребления.

Эффективная аппаратная реализация мультиконстантного умножителя

Мультиконстантный умножитель (англ. multi-constant multiplier или MCM) может быть реализован через набор сложений, вычитаний и сдвигов. В общем случае задача поиска минимального набора арифметических операций является NP-полной [2].

Рассмотрим сначала пример для одноконстантного умножителя. Пусть нам требуется построить умножитель вида $71x$, тогда можно записать формулу следующим образом:

$$70x = 1000110_2 x = x \ll 6 + x \ll 2 + x \ll 1$$

В случае многоконстантного умножителя появляется возможность оптимизаций за счёт выноса общих частей за скобку. Пусть требуется построить MCM блок с двумя коэффициентами $39x$ и $81x$:

Можно сделать промежуточные вычисления:

$$9x = x \ll 3 + x$$

И на основе них посчитать два требуемых коэффициента:

$$39x = x \ll 6 + 9x$$

$$81x = (9x) \ll 3 + 9x$$

В этом случае всё устройство построено на базе 3 сложений. Тогда как отдельная реализация каждого из одиночных умножителей потребовала бы минимум 4 сложения.

Алгоритмы для поиска аппаратной реализации мультиконстантных умножителей можно поделить на 4 большие категории:

- 1) Алгоритмы на базе цифровой перекодировки [3] – самая простая и быстрая группа методов, но с низкой эффективностью с точки зрения качества решения. Обычно используется на коэффициентах очень больших размерностей, где есть трудности у остальных методов;

- 2) Алгоритмы устранения общего подвыражения (CSE) [4-6] – развитие методов первой группы на базе поиска общих выражений в каждой из формул;
- 3) Алгоритмы на базе графов [7], RAG-n [8], и [9] – группа наиболее эффективных методов на базе итеративного построения графов;
- 4) Гибридные алгоритмы – алгоритмы сочетающие элементы первых трёх методов. Например [10].

Наиболее эффективным алгоритмом на данный момент является [11]. Для него так же есть доступный код с реализацией [12].

Объединение вычислительных блоков в общем случае приводит к уменьшению площади КИХ-фильтра и положительно коррелирует с уменьшением мощности. Блоков для объединения тем больше, чем больше число коэффициентов в фильтре и чем меньше они по разрядности (этот факт подтверждается экспериментами). Так же сильно влияет «похожесть» коэффициентов.

Модулярные КИХ-фильтры

Ранее было предложено несколько реализаций модулярных КИХ-фильтров на базе системы остаточных классов. Исследования показали, что некоторые архитектуры могут обеспечивать высокое быстродействие [13], в то время как другие архитектуры имеют преимущества с точки зрения энергоэффективности [14].

В работе предлагается применить модулярные принципы к проектированию транспонированных КИХ фильтров с постоянными коэффициентами (рис. 3). При построении модулярных мультиконстантных умножителей образуется еще больше общих частей за счет уменьшения коэффициентов фильтров. Таким образом появляется возможность более эффективной утилизации общих частей устройства, вследствие чего уменьшается общая потребляемая мощность. Однако в данном случае

необходимо особое внимание уделять выбору набора оснований, так прямой и обратный преобразователи из позиционной системы счисления могут существенно повлиять на энергопотребление.

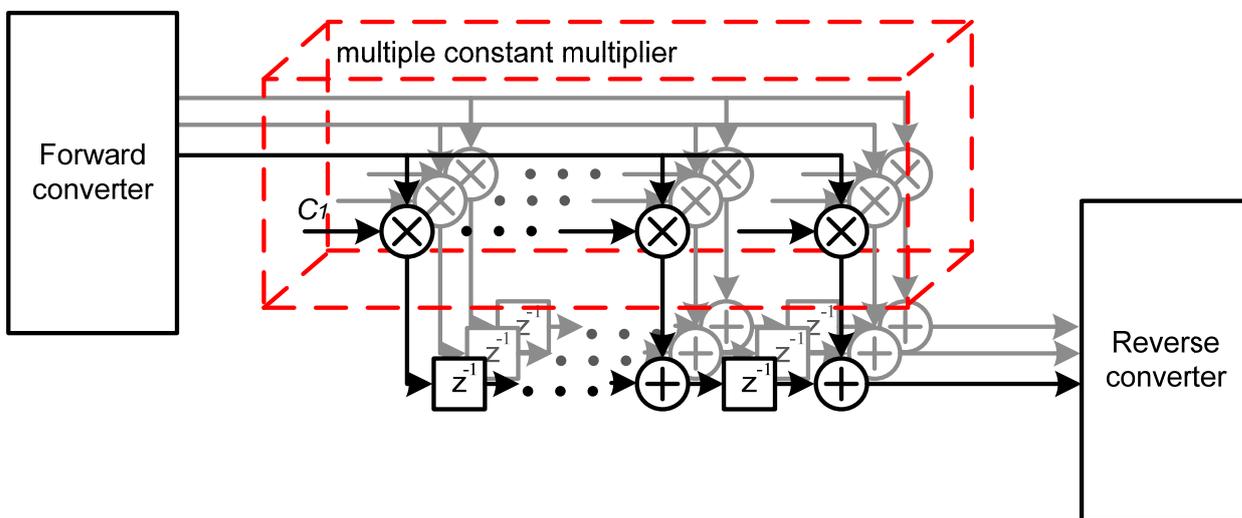


Рис. 3. – Структура модулярного КИХ-фильтра

Потребляемая мощность

Мощность, выделяемую в цепи можно подразделить на две широкие категории: статическую мощность и динамическую мощность.

Статическая мощность – это мощность, рассеиваемая затвором транзистора, когда он не переключается, то есть, когда он неактивен или статичен. Статическая мощность рассеивается несколькими способами. Наиболее вероятной причиной возникновения статической мощности является подпороговая утечка от истока до стока транзистора, которая вызвана пониженным пороговым напряжением, что предотвращает полное закрытие затвора транзистора. Статическая мощность также рассеивается, когда ток протекает между диффузионными слоями и подложкой. По этой причине статическую мощность часто называют мощностью утечки (leakage power).

Мощность утечки зависит от технологии производства (90/45/32 нм) и условий работы схемы (температура/напряжение). Рассчитанное значение

пропорционально площади схемы. Статическая или мощность утечки является единственной из составляющих частей полной мощности, модель которой в виде константной величины указывается производителем ASIC.

Power Compiler вычисляет полную мощность утечки дизайна путем суммирования токов утечки составных библиотечных ячеек в дизайне. Значения мощности утечки для отдельной взятой ячейки берется из Synopsys библиотеки (данные характеристики).

Динамическая мощность – это мощность выделяемая, когда цепь находится в активном состоянии. Цепь активна всякий раз, если напряжение в сети меняется из-за импульса на входе цепи. В виду того, что изменение напряжения во входной сети не влечет за собой обязательное изменение логического состояния на выходе, динамическая мощность может быть выделена, даже если выходная цепь не меняет своего логического состояния.

Динамическая мощность цепи есть сумма двух видов мощностей: коммутируемой (switching) и внутренней (internal).

Коммутируемая мощность выделяется при перезарядке выходных емкостей ячейки при переключении. Коммутируемая мощность отдельной ячейки есть функция полной емкости нагрузки на выходе ячейки и частоты логических переходов. Коммутирующая мощность всего дизайна вычисляется как сумма коммутирующих мощностей каждой отдельной ячейки.

Внутренняя мощность выделяется при изменении входов/выходов библиотечной ячейки. Внутренняя мощность выделяется, когда транзисторы NMOS and PMOS оба активны, создавая ток короткого замыкания. Значение закорткой или внутренней мощности пропорционально числу переключений цепи и зависит от емкости нагрузки на выходе цепи.

Во время расчета внутренней мощности, Power Compiler использует данные характеристики технической библиотеки. Внутренняя мощность

отдельно взятой ячейки есть сумма внутренних мощностей всех входов и выходов ячейки, как смоделировано в технологической библиотеке. Power Compiler производит расчеты, основываясь на переключательной активности, времени перехода и функциональности ячейки.

На рис. 4 изображен простой вентиль и показано, где рассеиваются статическая и динамическая мощности.

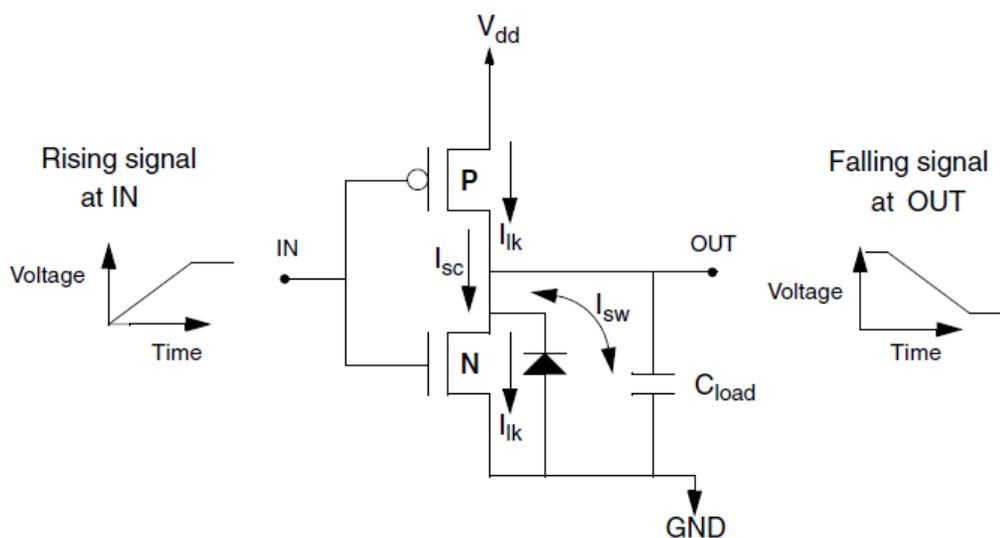


Рис. 4. – Компоненты выделенной мощности

I_{lk} – ток утечки; I_{sc} – ток короткого замыкания; I_{sw} – ток переключения.

Коммутируемая мощность обычно составляет от 70 до 90 процентов от всей рассеянной мощности. Внутренняя мощность, как правило, составляет от 10 до 30 процентов от рассеянной мощности. Утечка и статическая мощность составляют менее 1% от всей мощности рассеивания. Однако, утечка и статическая мощность могут легко стать основными источниками рассеивания мощности в дизайне, где цепь преимущественно неактивна, и это будет становится проблемой по мере уменьшения напряжения питания.

Экспериментальная часть

Экспериментальная часть посвящена исследованию энергопотребления транспонированного КИХ-фильтра с учетом минимизации модуля МСМ, описанного выше. Были рассмотрены 20 реализаций КИХ-фильтров с разной

разрядностью и набором коэффициентов, предложенные в научных публикациях. Каждая из реализаций КИХ-фильтра имеет своих авторов, что отражено в названии каждой из них (см. таблицу №1). Список фильтров и публикаций, в которых они были предложены, можно посмотреть в [15].

Таблица №1

Список КИХ-фильтров в эксперименте и их основные параметры

Реализация КИХ-фильтра	Разрядность коэффициентов	Количество коэффициентов	
		Всего	Уникальных
POTKONJAK96_4	11	4	4
GOODMAN77_E11	9	11	5
JAIN91_11	10	11	6
DEMPSTER04_15	13	5	5
VINOD03_26A	7	26	12
CHEN99_15	14	15	6
BULL91_32	8	32	15
LIM83_36	9	36	16
LIM83_37	9	37	16
CHENYAO01_28B	12	28	13
ROSA04_49	10	49	23
JOHANSSON08_30	11	30	29
SHI11_A	11	59	25
VINOD03_26B	15	26	13
KWENTUS97_47	16	47	14
SHI11_L2	11	63	30
SHAHEIN11_B	10	105	34
LIMAKT08_121	15	121	60
LIMPASKO99_121	15	121	58
LIM83_121	15	121	58

В таблице №1 разрядность, означает разрядность коэффициентов фильтра, так же указано общее количество коэффициентов и число уникальных, неповторяющихся коэффициентов. Например, для КИХ-фильтра CHEN99_15 коэффициенты фильтра: -32 0 223 0 -897 0 4034 6656

4034 0 -897 0 223 0 -32 из которых всего 6 уникальных. Цветом в таблице №1 отмечены наиболее удачные реализации в плане энергопотребления.

Каждой из реализаций КИХ-фильтра (таблица №1) поставлено в соответствие три Verilog-описания для проведения экспериментов: функциональное описание каноническую формы параллельного КИХ-фильтра или эталон; обычное транспонированное представление; транспонированное представление, минимизированное за счет уменьшения блока МСМ.

В тестбенче каждого Verilog нетлиста в качестве входных воздействий используется реальный звуковой сигнал, подгруженный в виде текстового файла, полученный средствами среды Matlab из звукового файла *.wav.

Для каждого Verilog-описания всех трёх версий КИХ-фильтра рассчитывается общее энергопотребление средствами Synopsys Power Compiler. Далее значения анализируются и сравниваются между собой. Процесс получения итоговых значений состоит из следующих этапов:

- средствами Synopsys Design Compiler синтезируются исходные Verilog-описания в нетлисты (на уровне вентилях) в базе ячеек свободно распространяемой стандартной библиотеки OpenNangate с технологическими нормами 90 нанометров;
- полученные нетлисты симулируются средствами Cadence NanoSim с тестбенчем реального звукового сигнала;
- полученный VCD файл с вейвформами конвертируется в SAIF файл, который содержит информацию по всем переключениям в схеме;
- SAIF используется в Synopsys Power Compiler наряду с синтезированными нетлистами, полученными на первом этапе.

В таблице №2 показаны результаты расчета энергопотребления для 20-ти различных КИХ-фильтров. Данные получены с помощью САПР Synopsys Power Compiler.

Таблица №2

Рассчитанное энергопотребление КИХ-фильтров

Реализация КИХ-фильтра	Общее энергопотребление, mW			Степень минимизации, %	Минимизированная vs Эталон, %
	Транспонированная форма	Минимизированная форма	Эталон		
POTKONJAK96_4	0.591	0.680	0.177	-15	284
GOODMAN77_E11	0.332	0.323	0.179	3	80
JAIN91_11	0.375	0.323	0.254	14	27
DEMPSTER04_15	1.223	1.788	0.358	-46	399
VINOD03_26A	0.524	0.459	0.304	12	51
CHEN99_15	1.489	0.894	0.776	40	15
BULL91_32	1.084	1.023	0.661	6	55
LIM83_36	1.304	1.068	0.888	18	20
LIM83_37	1.237	1.102	0.970	11	14
CHENYAO01_28B	1.840	1.311	1.333	29	-2
ROSA04_49	2.134	1.617	1.490	24	9
JOHANSSON08_30	3.129	2.785	1.421	11	96
SHI11_A	3.467	2.962	2.268	15	31
VINOD03_26B	5.956	6.684	3.297	-12	103
KWENTUS97_47	5.254	5.316	3.096	-1	72
SHI11_L2	3.526	2.903	2.304	18	26
SHAHEIN11_B	3.599	3.310	2.486	8	33
LIMAKT08_121	17.220	11.527	12.053	33	-4
LIMPASKO99_121	18.294	13.352	11.993	27	11
LIM83_121	18.628	13.194	12.494	29	6

Результаты отсортированы по возрастанию площади эталонных КИХ-фильтров, рассчитанных Design Compiler при синтезе. Таблица №2 отсортирована по 4-ому столбцу (эталон).

5-ый столбец (степень минимизации) показывает на сколько процентов понижается энергопотребление при минимизации транспонированной формы КИХ-фильтра (за счет минимизации блока MCM в Verilog-описании) по

сравнению с исходным значением (транспонированная форма). Энергопотребление транспонированной формы принимается за 100%. В рамках исследования те из реализаций, которые имеют наибольшие значения в этом столбце, считаются наиболее удачными.

6-ой столбец (он же последний) показывает, на сколько процентов минимизированная транспонированная форма КИХ-фильтра потребляет мощности больше, чем эталон (синтезированный на основе функционального описания КИХ-фильтра). Энергопотребление эталона принимается за 100%. Мы хотим увидеть наименьшие значения в этом столбце, чтобы считать наши реализации удачными.

Таким образом, мы выбрали группу реализаций (выделена цветом), удовлетворяющую нашим требованиям по энергозатратам.

На рис. 4 проиллюстрированы полученные результаты в виде гистограммы для выделенной группы реализаций КИХ-фильтра.



Рис. 5. – Полученные результаты для выделенной группы КИХ-фильтров

Таблица №3

Площадь КИХ-фильтров по результатам синтеза

Реализация КИХ-фильтра	Площадь после синтеза, um^2		
	Транспонированная форма	Минимизированная транспонированная форма	Эталон
POTKONJAK96_4	4666	3604	2050
GOODMAN77_E11	4174	3968	2252
JAIN91_11	4585	5068	2818
DEMPSTER04_15	7195	5954	3337
VINOD03_26A	9323	8381	5169
CHEN99_15	9741	7951	6341
BULL91_32	13665	13342	8445
LIM83_36	16801	16203	10114
LIM83_37	15637	16228	10147
CHENYAO01_28B	15527	14691	10798
ROSA04_49	22158	21898	14564
JOHANSSON08_30	27480	21069	14942
SHI11_A	32393	27918	20479
VINOD03_26B	29368	22665	20897
KWENTUS97_47	31148	27063	21122
SHI11_L2	32809	30968	21792
SHAHEIN11_B	43377	44169	26587
LIMAKT08_121	93587	83612	72898
LIMPASKO99_121	97080	85073	75011
LIM83_121	100739	84625	76133

Заключение

Анализируя полученные результаты, можно сделать вывод о том, что потребляемая мощность КИХ-фильтра прямо пропорциональна площади, занимаемой им на кристалле.

Кроме того, полученные результаты демонстрируют общую тенденцию к уменьшению площади транспонированного КИХ-фильтра и снижению его энергопотребления при удовлетворении следующих условий:

- КИХ фильтр имеет большое число повторяющихся коэффициентов (чем меньше уникальных, тем лучше результат);
- Разрядность коэффициентов КИХ фильтров мала (чем меньше значение, тем лучше результат).

Если эти свойства для КИХ-фильтра не выполняются, то для сокращения энергозатрат следует использовать каноническую форму фильтра (эталон).

Предполагается, что полученные закономерности справедливы и для КИХ-фильтров, спроектированных на основе модулярной арифметики. Для таких фильтров их коэффициенты будут удовлетворять условиям, описанным выше и приводящим к понижению потребляемой мощности. Также следует предположить, реализация КИХ-фильтра на базе рекурсивной модулярной арифметики [16], где коэффициенты ещё меньшей разрядности будет самой удачной с точки зрения энергозатрат.

Дальнейшая работа будет посвящена аппаратной реализации модулярных КИХ фильтров с минимизацией блока МСМ, а также вопросам выбора базисных модулей, эффективной реализации преобразователей и т. д.

Поддержка

Работа выполнена при поддержке гранта РФФИ 17-07-00404.

Литература

1. FIR filters tools // miniDSP URL: minidsp.com/applications/advanced-tools/fir-filter-tools (date of access: 07.12.2017).
2. P. Cappello and K. Steiglitz. Some complexity issues in digital signal processing. IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 32, No. 5, pp. 1037–1041, Oct. 1984.
3. Coleman, J. O. Cascaded coefficient number systems lead to FIR filters of striking computational efficiency. In Proc. 8th IEEE International Conference on Electronics, Circuits and Systems, Vol. 3, pp. 513-516, Sep. 2001.
4. R. Pasko, P. Schaumont, V. Derudder, S. Vernalde, and D. Durackova. A new algorithm for elimination of common subexpressions. IEEE Transactions on



Computer-Aided Design of Integrated Circuits and Systems, Vol. 18, No. 1, pp. 58-68, Jan. 1999.

5. Lefèvre, V. Multiplication by an integer constant: Lower bounds on the code length. In Proc. 5th Conference on Real Numbers and Computers, pp.131-146, 2003.

6. R.I. Hartley. Subexpression sharing in filters using canonic signed digit multipliers. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, Vol. 43, No. 10, pp. 677-688, Oct. 1996.

7. D.R. Bull and D.H. Horrocks. Primitive operator digital filters. IEE Proceedings G - Circuits, Devices and Systems, Vol. 138, No. 3, pp. 401–412, Jun. 1991.

8. A.G. Dempster and M.D. Macleod. Use of minimum-adder multiplier blocks in FIR digital filters. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, Vol. 42, No. 9, pp. 569–577, Sep. 1995.

9. R.L. Bernstein. Multiplication by integer constants. Software – Practice and Experience, Vol. 16, No. 7, pp. 641–652, 1986.

10. H. Choo, K. Muhammad, and K. Roy. Complexity reduction of digital filters using shift inclusive differential coefficients. IEEE Transactions on Signal Processing, Vol. 52, No. 6, pp. 1760–1772, Jun. 2004.

11. Y. Voronenko and M. Püschel. Multiplierless multiple constant multiplication. ACM Transactions on Algorithms (TALG), Vol. 3, No. 2, pp. 1–39, May 2007.

12. Spiral Generated Transform Libraries // Spiral URL: spiral.net/codegenerator.html (date of access: 07.12.2017).

13. V.M. Amerbaev, R.A. Soloviev, and D.V. Telpukhov. Hardware implementation of FIR filter based on number-theoretic fast fourier transform in Residue Number System. Open Engineering Sciences Journal, Vol. 1, No. 1, pp. 1-6, 2014.

14. P. Patronik, K. Berezowski, S. Piestrak, J. Biernat, and A. Shrivastava. Fast and energy-efficient constant-coefficient FIR filters using residue number system. In Proceedings of the 17th IEEE/ACM International Symposium on Low Power Electronics and Design, pp. 385-390, Aug. 2011.

15. FIR Filter data found in the literature // FIRsuite URL: firsuite.net/FIR/FromPublication (date of access: 07.12.2017).

16. Соловьев Р.А., Тельпухов Д.В. Методика выбора базисных оснований для рекурсивной модулярной арифметики // Вычислительные технологии. 2014. том 19, №4. С. 99-106.

17. Балака Е.С., Михмель А.С., Северцев В.Н., Гулякович Г.Н. Система синхронизации в микросхемах с высокой степенью интеграции // Инженерный вестник Дона, 2017, №2 URL: ivdon.ru/ru/magazine/archive/n2y2017/4115/.

18. Тельпухов Д.В., Рухлов В.С., Рухлов И.С. Исследование и разработка методов оценки сбоеустойчивости комбинационных схем, реализованных в базисе ПЛИС // Инженерный вестник Дона, 2016, №1 URL: ivdon.ru/ru/magazine/archive/n1y2016/3504/.

19. B. Chen and I. Nedelchev. Power Compiler: A Gate Level Power Optimization and Synthesis System. In Proc. IEEE International Conference on Computer Design: VLSI in Computers and Processors, pp. 74-79, Oct. 1997.

20. Synopsys Power Compiler User Guide. Version D-2010.03-SP2. June 2010. Synopsys, 2010. 453 p.

References

1. FIR filters tools. Date Views 07/12/2017. URL: minidsp.com/applications/advanced-tools/fir-filter-tools.

2. P. Cappello and K. Steiglitz. Some complexity issues in digital signal processing. IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 32, No. 5, pp. 1037–1041, Oct. 1984.

3. Coleman, J. O. Cascaded coefficient number systems lead to FIR filters of striking computational efficiency. In Proc. 8th IEEE International Conference on Electronics, Circuits and Systems, Vol. 3, pp. 513-516, Sep. 2001.
 4. R. Pasko, P. Schaumont, V. Derudder, S. Vernalde, and D. Durackova. A new algorithm for elimination of common subexpressions. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 18, No. 1, pp. 58-68, Jan. 1999.
 5. Lefèvre, V. Multiplication by an integer constant: Lower bounds on the code length. In Proc. 5th Conference on Real Numbers and Computers, pp.131-146, 2003.
 6. R.I. Hartley. Subexpression sharing in filters using canonic signed digit multipliers. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, Vol. 43, No. 10, pp. 677-688, Oct. 1996.
 7. D.R. Bull and D.H. Horrocks. Primitive operator digital filters. IEE Proceedings G - Circuits, Devices and Systems, Vol. 138, No. 3, pp. 401–412, Jun. 1991.
 8. A.G. Dempster and M.D. Macleod. Use of minimum-adder multiplier blocks in FIR digital filters. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, Vol. 42, No. 9, pp. 569–577, Sep. 1995.
 9. R.L. Bernstein. Multiplication by integer constants. Software – Practice and Experience, Vol. 16, No. 7, pp. 641–652, 1986.
 10. H. Choo, K. Muhammad, and K. Roy. Complexity reduction of digital filters using shift inclusive differential coefficients. IEEE Transactions on Signal Processing, Vol. 52, No. 6, pp. 1760–1772, Jun. 2004.
 11. Y. Voronenko and M. Püschel. Multiplierless multiple constant multiplication. ACM Transactions on Algorithms (TALG), Vol. 3, No. 2, pp. 1–39, May 2007.
-



12. Spiral Generated Transform Libraries. Date Views 07/12/2017 URL: spiral.net/codegenerator.html.

13. V.M. Amerbaev, R.A. Soloviev, and D.V. Telpukhov. Open Engineering Sciences Journal, Vol. 1, No. 1, pp. 1-6, 2014.

14. P. Patronik, K. Berezowski, S. Piestrak, J. Biernat, and A. Shrivastava. In Proceedings of the 17th IEEE/ACM International Symposium on Low Power Electronics and Design, pp. 385-390, Aug. 2011.

15. FIR Filter data found in the literature. Date Views 07/12/2017 URL: firsuite.net/FIR/FromPublication.

16. Solovyev R.A., Telpukhov D.V. Vychislitel'nye tekhnologii. 2014. vol. 19, no. 4, pp. 99-106.

17. E.S. Balaka, A.S. Mikhmel, V.N. Severtsev, G.N. Gulyakovich Inzhenernyj vestnik Dona (Rus), 2017, №2. URL: ivdon.ru/ru/magazine/archive/n2y2017/4115/.

18. D.V. Telpukhov, V.S. Rukhlov, I.S. Rukhlov Inzhenernyj vestnik Dona (Rus), 2016, №1. URL: ivdon.ru/ru/magazine/archive/n1y2016/3504/.

19. B. Chen and I. Nedelchev. Power Compiler: A Gate Level Power Optimization and Synthesis System. In Proc. IEEE International Conference on Computer Design: VLSI in Computers and Processors, pp. 74-79, Oct. 1997.

20. Synopsys Power Compiler User Guide. Version D-2010.03-SP2. June 2010. Synopsys, 2010. 453 p.