

Схема работы программы

Основная программа формирует сигнал, подготавливает материнский вейвлет с помощью модуля GSL и выполняет параллельноевейвлет-преобразование с помощью модуля cudaDwt.

Процедура вейвлет-преобразования основана на взаимодействии следующих модулей:

- 1) CudaDwt – исполняемая статическая/динамическая библиотека;
- 2) GSL – свободно распространяемая библиотека для научных вычислений.

Для реализации алгоритма вейвлет-фильтрации в структуре параллельных вычислений необходимо использовать требуемые *CUDA* расширения языка Си, переписать исходный код последовательной реализации, убрать конструкцию цикла, и сделать вычисление адреса текущего элемента во входной последовательности через уникальные идентификаторы процессора потока [2]. Под задачу прямого преобразования необходимо производить расчет числа процессоров (число процессоров равно числу входных элементов).

При пороговой обработке для каждого элемента данных можно запустить свой поток обработки.

Возможен вариант реализации восстановления данных при обратном ДВП. Реализация данного алгоритма в параллельной структуре приведена в работе [2].

#### Заключение

В рамках статьи приведен пример разработки УЦОС ТВТСО на основе анализа основных методов подавления шума на изображениях и рассмотрен метод вейвлет-фильтрации, как наиболее полно отвечающий задачам, решаемым разрабатываемой ТВТСО. Представлен вариант реализации вейвлет-фильтра средствами *GPGPU*.

#### Список литературы

1. Проблема подавления шума на изображениях и видео и различные подходы к ее решению. Компьютерная Графика и Мультимедиа. Режим доступа <http://cgm.computergraphics.ru/content/view/74, свободный> (дата обращения 11.04.2014г.). Заголовок с экрана.
2. Скоморохов Е.Ю. Эффективность вейвлет фильтрации сигнала на *GPGPU*. Режим доступа <http://www.bysolo.ru>, свободный (дата обращения 20.04.2014г.). Заголовок с экрана.

#### ПЕРСПЕКТИВЫ РЕАЛИЗАЦИИ АЛГОРИТМОВ ЦИФРОВОЙ ФИЛЬТРАЦИИ

Уваров С.А., Борисова С.Н.

Пензенский государственный технологический университет, Пенза, Россия, e-mail: romi\_s@list.ru

С проектированием цифровых фильтров (ЦФ) так или иначе приходилось сталкиваться большинству разработчиков современных систем связи и управления. Проектирование ЦФ в представлении большинства разработчиков прочно ассоциируется с процедурой синтеза структуры ЦФ частотной селекции, причем наиболее простой его разновидности, предназначенной для выделения (подавления) одной априорно заданной полосы частот. Однако в последние десятилетия появились другие разновидности ЦФ – многокаскадные структуры с прореживанием по времени и по частоте, фильтры Винера и Калмана, робастные и адаптивные. Кроме того наблюдается тенденция к расширению сферы применения оптимальных и адаптивных ЦФ. Таким образом, необхо-

дима формализация относительно новых и сложных принципов ЦОС в виде т.н. мегафункций, реализованных на том или ином языке описания аппаратуры (AHDL, VHDL и т.п.).

Ниже приведена классификация (ЦФ), в основу которой положен функциональный принцип и перспективные задачи в области проектирования указанных фильтров.

ЦФ частотной селекции. Это наиболее известный, хорошо изученный и апробированный на практике тип ЦФ. С алгоритмической точки зрения ЦФ частотной селекции могут рассматриваться как результат перехода на новую элементную базу традиционных аналоговых фильтров частотной селекции. Перспективная задача в области проектирования: разработка и доведение до мегафункций новых подходов к проектированию ЦФ частотной селекции, альтернативных существующим (на цифровых процессорах обработки сигналов (ЦПОС) или в форме КИХ-фильтра на ПЛИС), возможно, с использованием БИХ-фильтров.

Оптимальные (квазиоптимальные) ЦФ. Этот тип фильтров применяется тогда, когда требуется оценить те или иные физические величины, характеризующие состояние системы, подверженной случайным возмущениям. Современная тенденция - использование достижений теории оптимальной фильтрации и реализация устройств, минимизирующих средний квадрат ошибки оценивания. Они подразделяются на линейные и нелинейные в зависимости от того, какими уравнениями описывается состояние системы. Перспективная задача в области проектирования: создание ориентированной на пользователя библиотеки алгоритмов, реализуемых оптимальными (квазиоптимальными) ЦФ, с указанием входных и выходных параметров, реализуемых операций и пригодной для непосредственного использования при моделировании сложных систем.

Адаптивные ЦФ. Сущность адаптивной цифровой фильтрации состоит в следующем: для обработки входного сигнала (обычно адаптивные ЦФ строят одноканальными) используется обычный КИХ-фильтр, однако импульсная характеристика (ИХ) этого фильтра не остается раз и навсегда заданной, как это было при рассмотрении ЦФ частотной селекции. Она также не изменяется по априорно заданному закону, как это происходит в ЦФ Калмана. ИХ корректируется с поступлением каждого нового отсчета таким образом, чтобы свести к минимуму среднеквадратическую ошибку фильтрации на данном шаге. Под адаптивным алгоритмом понимается рекуррентная процедура пересчета вектора отсчетов ИХ на предыдущем шаге в вектор "новых" отсчетов ИХ для следующего шага. Проблема для разработчика состоит в том, что априорно устойчивые адаптивные алгоритмы, как правило, недопустимо сложны в реализации, а более простые (как МНК - алгоритм) могут расходиться. Одним из возможных методов преодоления указанных трудностей является каскадирование БИХ-фильтров, построенных на основе решетчатых структур.

Как видно из предшествующего изложения, без широкого применения средств САПР об успешной разработке ЦФ не может быть и речи. Поэтому одной из главных задач становится создание библиотек, реализующих те или иные решения.

Многие ЦФ достаточно просто реализовать в виде КИХ-фильтра. С появлением БИС семейства FLEX8000 и FLEX10K фирмы ALTERA появилась возможность создания гибких КИХ-фильтров высокого порядка с производительностью более 100 MSPS.

Рассмотрим особенности реализации КИХ-фильтров на базе ПЛИС семейств FLEX8000 и FLEX10K с учетом специфики их архитектуры на

примере КИХ-фильтра 8-го порядка. На рис. 1 а представлен параллельный векторный перемножитель, введены следующие обозначения:  $x_n$  – сигнал на выходе  $n$ -го регистра,  $h_n$  – коэффициент СФ,  $y_n$  – выходной сигнал. Сигнал на выходе фильтра будет иметь вид:

$$y_n = \sum_{k=1}^8 x_n h_k$$

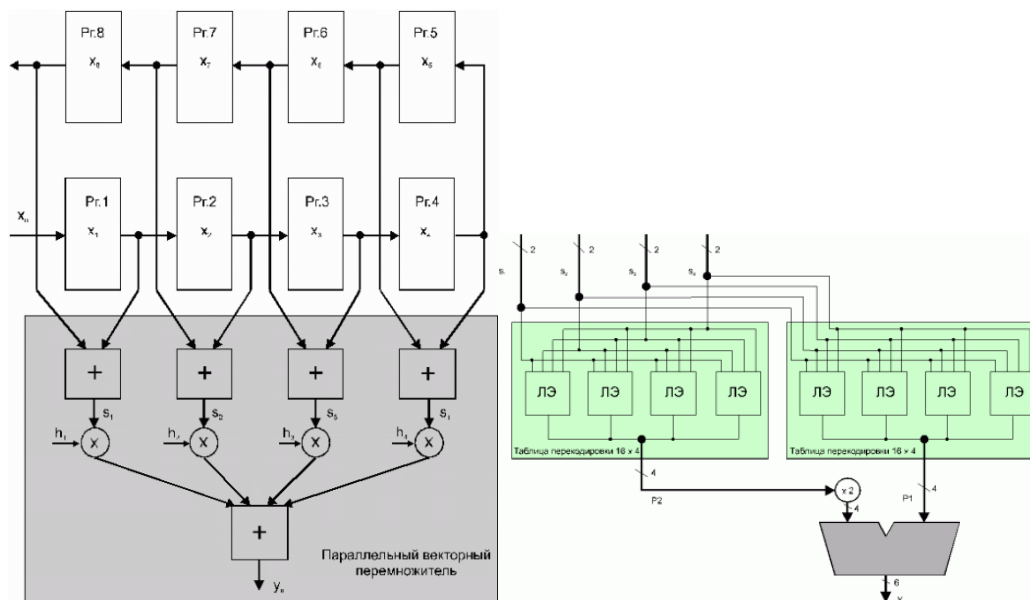


Рис. 1. Параллельный векторный перемножитель

При использовании ТП операция перемножения выполняется параллельно. Структура четырехразрядного ПВП представлена на рисунке 1,б. Основным достоинством параллельной архитектуры построения фильтров является высокая производительность. Однако, если число задействованных ЛЭ является критичным, предпочтительнее строить фильтр по последовательной или комбинированной архитектуре. Комбинированная архитектура является компромиссом между экономичностью и производительностью. В этом случае применяются как последовательные, так и параллельные регистры. В результате распараллеливания вычислений обеспечивается несколько большая производительность, чем у последовательного фильтра.

Для проектирования фильтров необходим набор специализированных программных средств, позволяющих синтезировать требуемую системную функцию фильтра, провести моделирование его работы, а также библиотеки параметризуемых мегафункций, содержащих реализации типовых структур ЦФ. Фирма ALTERA предоставляет в составе Altera DSP Design KIT программный продукт Firgen. В функции программы Firgen входит моделирование КИХ-фильтра с заданными коэффициентами, а также генерация файлов, предназначенных для реализации фильтра штатными средствами пакета MAX+plus II. Для построения отклика фильтра, полученного в результате моделирования средствами Firgen можно использовать либо пакет Microsoft Excel, либо входящий в состав Altera DSP Design KIT продукт GNUplot. Схематический процесс разработки фильтра представлен на рис. 2.

Рассмотрев основные алгоритмы цифровой фильтрации и пути их реализации на ПЛИС, можно сделать следующие выводы: разработчики осознают необходимость создания библиотек параметризуемых мегафункций различных функциональных узлов, особенно устройств цифровой обработки сигналов. Определен-

Для обеспечения линейности фазовой характеристики КИХ-фильтра коэффициенты  $h_n$  СФ фильтра выбираются симметричными относительно центральной величины. Такое построение позволяет сократить число перемножителей. В качестве параллельного векторного перемножителя (ПВП) удобно использовать таблицу перекодировок (ТП, LUT, Look-up table), входящую в состав логического элемента (ЛЭ) ПЛИС.

ные шаги в этом направлении предпринимает фирма ALTERA. Однако, на рынке до сих пор очень мало предложений в области нелинейных, оптимальных и большинства типов адаптивных структур. Между тем, существует огромная потребность в реализации известных и хорошо обоснованных теоретически алгоритмов.



Рис. 2

### Список литературы

1. Перспективы реализации алгоритмов цифровой фильтрации на основе ПЛИС фирмы "Altera" // <http://www.dsol.ru/stud/book5/df.html>.
2. Шипулин С.Н., Храпов В.Ю. Особенности проектирования цифровых схем на ПЛИС - Chip News, 1996, № 5, С. 40.
3. Шипулин С.Н., Храпов В.Ю. ПЛИС фирмы Altera // Chip News, 1996, № 2, С. 44.

### МЕТОДЫ СПЕЦИАЛЬНОГО КОДИРОВАНИЯ ДАННЫХ С ОГРАНИЧЕНИЕМ ДЛИНЫ ПОЛЯ ЗАПИСИ

Хохлов Д.В., Шмокин М.Н.

Пензенский государственный технологический университет  
Пенза, Россия, e-mail: romi\_s@list.ru

#### Введение

При воспроизведении данных, записанных на носитель записи, сигналы данных как в ВЗУ на оптических дисках, так и в ВЗУ на магнитных дисках претерпевают искажения, обусловленные ограниченной разрешающей способностью системы головки – диск. Критерием этой разрешающей способности является длительность одиночного импульса воспроизведения, которая обычно превышает длительность синхротакта, в результате чего происходит межсимвольная интерференция (МСИ) и суммарный (результатирующий) сигнал данных имеет амплитудные и временные искажения.

#### Причины использования кодов с ограничением длины поля записи

На рис. 1 приведена записанная кодограмма данных (1), а на (3) и (5) показаны сигналы данных воспроизведения в ВЗУ на оптических и магнитных дисках, соответственно.

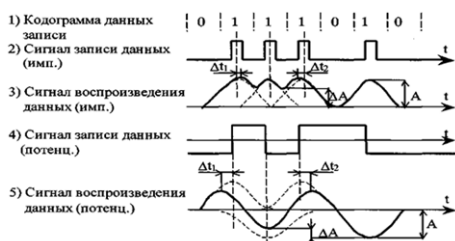


Рис. 1. Кодограммы данных под влиянием МСИ

Искажения сигнала воспроизведения за счет МСИ состоят в непостоянстве амплитуды сигнала и сдвиге во времени пиков сигнала от их номинальной позиции, соответствующей позиции «единичного» элемента сигнала записи.

При воспроизведении данных, записанных на носитель, решается задача идентификации каждого элемента («единичного» и «нулевого»), что реализуется путем потактной конъюнкции сигнала данных сигналом специальной синхросерии воспроизведения, формируемой под управлением сигналов данных воспроизведения. Эта синхросерия создается синхрогенератором, управляемым «единичными» импульсами сигнала воспроизведения. Такое управление обеспечивает регулирование частоты синхросерии, отслеживающее текущие изменения скорости движения носителя при воспроизведении. Это необходимо для минимизации сдвига во времени сигналов синхросерии относительно сигналов данных. Такой сдвиг означает как бы уменьшение эффективного временного допуска на разрешенное расположение во времени каждого элементарного сигнала данных по сравнению с номинальным значением этого допуска, равным синхротакту  $T_c$ . Результатом такого уменьшения временного допуска является уменьшение достоверности воспроизведения данных, то есть повышение вероятности сбоя или ошибки в данных.

Для повышения точности отслеживания частоты синхросерии за скоростью носителя данных подстройка синхрогенератора посредством «единичных» импульсов должна выполняться достаточно часто, чтобы минимизировать накопленный «уход» частоты относительно текущей средней частоты данных воспроизведения. Обычный двоичный код имеет равную вероятность появления символов «1» или «0» в текущем такте и допускает бесконечно большие «нулевые» последовательности символов, в пределах которых нет подстройки частоты генератора. Такой код не обладает свойствами самосинхронизации. В связи с этим для представления данных на подвижном носителе в ВЗУ создают специальные каналные коды, которые гарантируют появление символа «1» через определенное количество символов «0» и поэтому обладают свойствами самосинхронизации.

Существует две причины, по которым необходимо это техническое оснащение: во-первых, синхронизируемая (синхронизирующая) схема использует поток импульсов, благодаря переходам чередующихся магнитных полярностей на диске, чтобы поддерживать правильное время связывания-удержания (*track-and-hold*), ADC и детектора. В то же время, если импульсы долгое время отсутствуют, т.е. длинная последовательность нулей (запись без возврата к нулю с инверсией – *NRZrecording*), канал двусторонней связи не получит достаточно информации, чтобы обеспечить надлежащую синхронизацию (выравнивание). Вследствие этого, временной промежуток между двумя единицами (1) должен быть ограничен. Во-вторых, когда два бита записываются слишком близко друг к другу, взаимные помехи между двумя противоположными импульсами будут снижать соотношение «сигнал-шум» (*SNR*) при считывании только что записанной информации (эхо считывании). Следовательно, следует использовать постоянное кодирование, чтобы гарантировать расстояние (интервал) между двумя переходами расстояние (интервал), достаточно большой, чтобы избежать межсимвольных помех (*ISI*); поскольку это расстояние (интервал), которое обычно определяется от средней величины, техническое оснащение в состоянии затем увеличить плотность записи. Выравнивание по частичному отклику применяется сейчас для борьбы с проблемой межсимвольных помех (*ISI*), но кодирование с ограничением длины поля записи (*RLL*) широко используется по вышеприведенным причинам [1].

#### Особенности каналных самосинхронизирующихся кодов

Кодирование с ограничением длины поля записи (*RLL*) обычно классифицируют по  $(d, k)$ : показатель  $d$  отражает минимальное количество нулей между двумя единицами (1). Показатель  $k$  ограничивает максимальное количество нулей между двумя единицами (1). С другой точки зрения,  $d$  контролирует высокочастотные составляющие в связанных сигналах токов записи, чтобы сократить межсимвольные помехи (*ISI*).  $k$  же, напротив, оказывает доминирующее влияние на низкочастотные составляющие, чтобы обеспечить информацию о частоте для канала тактовой синхронизации. Рис. 2 показывает диаграмму переходов для типовых  $(d, k)$  кодов:

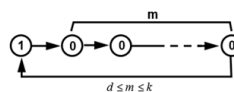


Рис. 2. Диаграмма переходов  $(d, k)$  кодов

Параметр  $d$  может принимать значения из натурального ряда целых чисел, включая 0, то есть