

УДК 004.043

**РЕАЛИЗАЦИЯ АЛГОРИТМА ШУМОПОДАВЛЕНИЯ В РЕЧЕВОМ ТРАКТЕ
СИСТЕМ МОБИЛЬНОЙ СВЯЗИ НА БАЗЕ СБИС**

С.О. Чураев, А.Д. Адамова, Т.Т. Палташев

Приведен анализ вопросов аппаратной реализации комплексных алгоритмов цифровой фильтрации для речевого канала систем сотовой связи и последующей их коммерциализации на основе полузаказных СБИС и ПЛИС. Использован алгоритм адаптивного шумоподавления с применением прямого/обратного преобразования Фурье и фильтрации речевого сигнала по методу Винера.

Ключевые слова: фильтр Винера, временные домены, частотные домены, тактовые частоты, двухпортовая память, конечный автомат.

Введение

В данной работе отражены актуальные вопросы аппаратной реализации комплексных алгоритмов цифровой фильтрации в заказных сверхбольших интегральных схемах (СБИС) и программируемых логических интегральных схемах (ПЛИС) [1–5]. Рассматривается переход от алгоритмов, реализованных программно на языке С и адаптированных под микропроцессорную платформу, к проектированию СБИС на языке Verilog [6–8]. Применен алгоритм адаптивного подавления внешнего акустического шума, проникающего в канал связи вместе с речью человека через микрофон и не несущего полезной информации [1–5]. Для решения этой задачи использовано прямое/обратное преобразование Фурье и винеровская фильтрация речевого сигнала. Конечной целью является создание законченного RTL кода на языке Verilog для последующего изготовления специализированного ядра заказной СБИС шумоподавления и установки в телефоны компаний-производителей систем мобильной связи. Проектирование СБИС обусловлено требованиями увеличения скорости обработки данных и снижения энергопотребления подобных устройств. Необходимость выполнения данного проекта также продиктована требованием заказчиков снижения цены ключевых компонентов продукта в 5–10 раз. Далее рассмотрены важнейшие моменты аппаратной реализации [7].

Системное аппаратное решение

Для комплексного тестирования системы разработано устройство, структурная схема которого приведена на рис. 1. В его состав входят: аудиокодек для аналогово-цифрового и цифро-аналогового преобразования сигнала и формирования цифрового потока для ПЛИС (audio codec); специальный конфигурационно-тестовый порт RS-232 для связи с компьютером (RS-232); ПЛИС для формирования кадров, выполнения прямого/обратного преобразования Фурье и проверки винеровского алгоритма фильтрации (FPGA Virtex-4); персональный компьютер для обеспечения подачи тестового входного сигнала с шумом и записи выходного очищенного сигнала (PC) [6, 7].

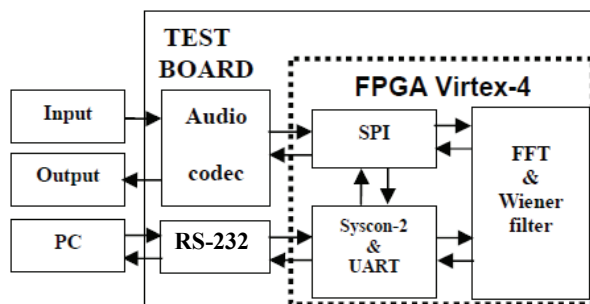


Рис. 1. Структурная схема аппаратного решения устройства фильтрации

Реализация алгоритма в ПЛИС

Входной речевой сигнал, оцифрованный с частотой дискретизации 8 кГц, разделяется на два кадра по 32 16-битных отсчета в каждом, которые формируются во входной двухпортовой памяти размером 64×16 бит. После того, как формируется первый кадр из 32 значений, начинается формирование второго кадра с адреса 33, и далее до 64. Одновременно через второй порт, но на значительно более высокой скорости, происходит считывание данных и их преобразование модулями Фурье и винеровской фильтрации. По окончании работы эти модули записывают информацию в выходную двухпортовую память, где обработанные данные будут в дальнейшем прочитаны низкоскоростным устройством связи с помощью аудиокодека. Таким образом, к моменту формирования второго кадра вся необходимая работа по фильтрации сигнала первого кадра будет завершена. Соответственно, по сигналу готовности заполнения данных второго кадра (с 33 по 64) будет проведена аналогичная работа, по мере того как заполняется первая половина памяти с 1 по 32 адрес, и т.д.

Особенности передачи данных при переходе через временные домены внутри ПЛИС

Попытка реализовать схему, используя классические методы цифрового преобразования, натолкнулась на противоречия между расчетом тактовой частоты ядра, его размером, скоростью работы и энергопотреблением системы [6, 7]. При реализации алгоритма в одном временном домене выяснилось, что скорость работы ядра недостаточна для выполнения всех шагов. По этой причине схема была разделена на несколько временных доменов, что позволило увеличить производительность каждого модуля по отдельности, а также снизить энергопотребление за счет снижения тактовой частоты модулей.

Говоря об особенностях реализации алгоритма, следует отметить ситуацию, при которой модули, работая синхронно, должны обеспечить передачу данных между соседними модулями, работающими в разных временных доменах [8]. Для синхронизации передачи данных реализовано три уровня контроля и подтверждения передачи данных:

1. асинхронный;
2. синхронный протокол обмена 2-х модулей типа «handshake»;
3. системный контроллер кадров.

Асинхронный контроль подразумевает предотвращение метастабильных состояний путем введения цепочек триггеров в цепях передачи данных между модулями при выполнении переходов через границы тактовых доменов. Синхронный протокол обеспечивает контроль состояний двух соседних модулей и отрабатывает протоколирование сигналов готовности, приема, передачи, запроса, а также сигнализации о переходе в запрещенные состояния.

Третий уровень контроля предоставляет широкие возможности контроля и управления объектом: отслеживание номера кадра и номера процесса (шага), подсчет контрольной суммы проходящих данных, обеспечение режима паузы, сброса и пошагового выполнения, обеспечение различных режимов передачи данных в обход некоторых модулей (режим BYPASS).

Двухпортовая память для реализации высокоскоростного алгоритма фильтрации

Как уже отмечалось, ключевым элементом схемы служит двухпортовая межмодульная память (рис. 2). Применение двухпортовой памяти оправдано удобством независимой адресации чтения или записи данных разных ее областей. Таким образом, появляется возможность последовательно передавать обрабатываемые данные с высокой скоростью через все модули ПЛИС/СБИС. Процесс передачи данных от одного модуля к другому через двухпортовую память накладывает некоторые ограничения на ее работу. Предусматривается защита от чтения данных, которые еще не записались в память (конфликт RAW). Аналогично, не проводится запись в область памяти, которая читается другим модулем (конфликт WAR). Таким образом, при обмене данными разработан и реализован синхронный протокол обмена двух модулей типа «handshake» [6].

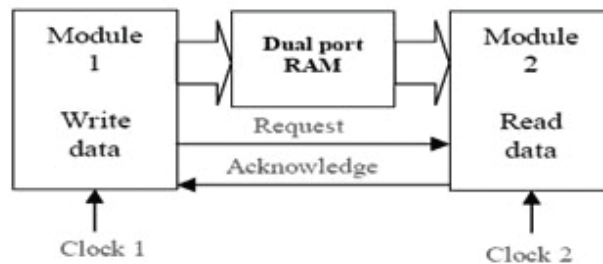


Рис. 2. Схема передачи данных через двухпортовую память

Протокол гарантированной передачи данных между временными доменами

Разработан протокол обмена данными с использованием двухпортовой памяти для межмодульного интерфейса в различных временных (частотных) доменах. В системе реализуется конечный автомат состояний для проверки этапов выполнения передачи данных [8]. На рис. 3 представлены временные диаграммы передачи данных между временными доменами.

Первый модуль (ведущий) работает на тактовой частоте Clock 1 и формирует сигнал запроса на чтение (Request). Второй модуль (ведомый), работающий на частоте Clock 2, формирует сигнал ответа (Acknowledge) и ожидает снятия сигнала от первого модуля (Request). При ответе первого модуля и установке в ноль сигнала Request второй модуль выполняет чтение и обработку данных из двухпортовой памяти, а затем переводит сигнал Acknowledge в начальное состояние. В итоге два модуля переходят в исходное состояние. Таким образом, можно выделить пять основных состояний:

1. запрос на передачу данных ведущего модуля;
2. подтверждение запроса от ведомого модуля;
3. снятие запроса;
4. чтение данных ведомым модулем из второго порта двухпортовой RAM;
5. снятие подтверждения и переход обоих модулей в исходное состояние.

Таким образом, асинхронные данные могут быть гарантированно переданы от одного модуля к другому.

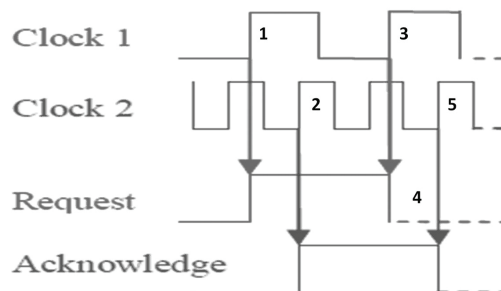


Рис. 3. Временные диаграммы передачи данных между временными доменами

Оптимизация скорости работы и размера конвейера

Под конвейером будем понимать последовательность элементов памяти, разделенных логическими устройствами (рис. 4). По мере того, как решается проблема недостаточной производительности СБИС, возникает проблема оптимизации конвейера [8]. Максимально возможная тактовая частота в одном временном домене будет определяться наибольшей возможной тактовой частотой самого медленного из модулей. Тактовая частота, на которой может работать модуль, определяется временем прохожде-

ния сигнала от элементов памяти через логические элементы к последующим элементам памяти. Если тактовые частоты сильно различны, то будут происходить существенные потери производительности всего конвейера. С целью выравнивания скоростей работы отдельных частей конвейера предложено перераспределять часть логических элементов между его различными участками. Таким образом, общая задержка обработки данных в одном временном домене будет иметь тенденцию к снижению.

Одновременно выполнены не только оптимизация и перенос логических элементов внутри одного домена, но также и перенос функциональности между различными временными доменами. Это обусловлено требованиями уменьшения потерь скорости при передаче данных между различными временными доменами через двухпортовую память и связано с особенностями структуры ПЛИС при синтезе и размещении элементов при выборе оптимального соотношения количества логических элементов и блочной памяти в ПЛИС.

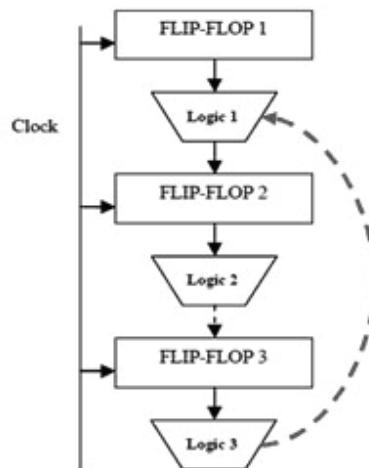


Рис. 4. Конвейер

Синхронизация асинхронных событий. Решение проблемы метастабильности в ПЛИС

Многие системы, предназначенные для совместной работы, имеют асинхронные входные сигналы, которые должны быть синхронизированы в различных временных доменах [8]. В общем случае изменение сигналов может происходить в любое время. Обычный триггер, получая асинхронный сигнал на вход, может перейти в метастабильное состояние, если есть нарушения во времени предустановки или удержания данных по отношению к тактовому сигналу (setup&hold). В предложенном проекте это состояние предотвращается посредством использования последовательного соединения двух и более триггеров (рис. 5).

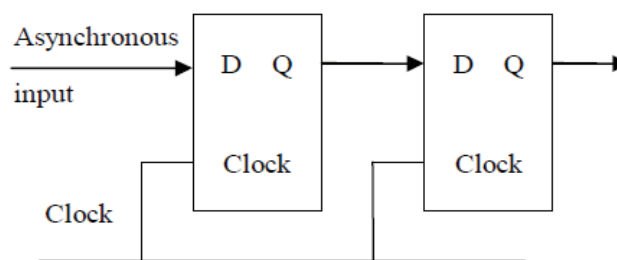


Рис. 5. Синхронизация передачи асинхронных данных с помощью двух триггеров

Рекомендации по аппаратной реализации комплексных алгоритмов фильтрации

В процессе синтеза и размещения элементов особое внимание уделено обнаружению и удалению избыточной логики, обнаружению комбинационных петель обратной связи, поиску состояний холостого хода или простоя, определения неиспользуемых (пустых) состояний конечных автоматов, определения и слияния эквивалентных состояний конечных автоматов [6]. Предложены правила проектирования архитектуры ПЛИС для реализации алгоритмов фильтрации:

- не использовать задержки в RTL коде (только на поведенческом верхнем уровне окружения);
- разрабатывать полностью синхронный дизайн в границах одного тактового домена;
- выполнять рекомендации по переходу между временными доменами; контролировать все возможные этапы передачи данных; избегать метастабильности в схемах;

- использовать различные схемы самотестирования (BIST) для контроля качества, тестирования и работоспособности устройства;
- для быстрого обнаружения ошибок при отладке RTL кода применять высокореалистичную модель тестовой среды; использовать случайные и детерминированные задержки, вводить дрожание тактовых сигналов и сигналов данных; не рекомендуется использование триггеров, работающих по уровню (LATCH);
- спроектировать высокодетализированный командный файл синтеза с указанием всех уникальных имен регистров и связей из RTL кода.

Заключение

Предложенные в работе архитектурные решения реализованы в коммерческом проекте по разработке модуля СБИС для подавления шума в речевом тракте систем сотовой связи компаний-производителей мобильного оборудования Юго-Восточной Азии.

Обоснована необходимость использования двухпортовой памяти с целью увеличения производительности системы, определен базовый протокол обмена данными между двумя соседними модулями, работающими в разных временных доменах. Решены вопросы увеличения производительности ядра путем оптимизации структуры конвейера, а также переноса функциональности между разными временными доменами. Разработаны базовые рекомендации проектирования для сходных проектов и задач [6].

Выполнен синтез RTL кода и сравнительный анализ количества компонентов системы до и после оптимизации (таблица). Правильно выполненная реализация алгоритма на языке Verilog и последующая оптимизация схемы при синтезе позволяют экономить площадь кристалла, увеличивать быстродействие, уменьшать энергопотребление устройства, что актуально для работы мобильных устройств.

ДО ОПТИМИЗАЦИИ	ПОСЛЕ ОПТИМИЗАЦИИ
Selected Device : 4vlx25sf363-10	Selected Device : 4vlx25sf363-10
Number of Slices : 31429 out of 10752 292%	Number of Slices : 8053 out of 10752 74%
Number of Slice Flip Flops: 23509 out of 21504 109%	Number of Slice Flip Flops : 7071 out of 21504 32%
Number of 4 input LUTs : 55301 out of 21504 257%	Number of 4 input LUTs : 16474 out of 21504 76%
Number used as logic : 55250	Number used as logic : 16312
Number used as Shift registers: 37	Number used as Shift registers : 83
Number used as RAMs : 14	Number used as RAMs : 79
Number of IOs : 16	Number of IOs : 16
Number of bonded IOBs : 16 out of 240 6%	Number of bonded IOBs : 16 out of 240 6%
Number of FIFO16/RAMB16s : 3 out of 72 4%	Number of FIFO16/RAMB16s : 3 out of 72 4%
Number used as RAMB16s : 3	Number used as RAMB16s : 3
Number of GCLKs : 8 out of 32 25%	Number of GCLKs : 8 out of 32 25%

Таблица. Распределение ресурсов до и после оптимизации

Литература

1. Extrapolation, interpolation and smoothing of stationary time series / N. Wiener // New-York, 1949.
2. Advanced digital signal processing and noise reduction / S.V. Vaseghi // 3-rd ed. – Wiley. – 2006.
3. Обробка сигналів. Цифрова обробка дискретизованих сигналів / Е. Шрюфер. – Київ: – Либідь, 1992. – 294 с.
4. Сергиенко А.Б. Алгоритмы адаптивной фильтрации: особенности реализации в Matlab [Электронный ресурс]. – Режим доступа: <http://images.nature.web.ru/nature/2003/04/11/0001193683/06.pdf>, своб.
5. Курсы лекций: Часть 1. Цифровая обработка сигналов. Национальный технический университет Украины (Киевский политехнический институт) [Электронный ресурс]. – Режим доступа: <http://aprodeus.narod.ru/teaching.htm>, своб.
6. FPGA FFT Implementation / S.O. Churayev, B.T. Matkarimov // Proceedings of IEEE East-West Design & Test Symposium (EWDTS'09), September 18–21, 2009. – Moscow. – 2009. – P. 96–98.
7. Churayev S.O., Matkarimov B.T. FPGA implementation of noise reduction system for mobile devices, Korean Scientific & Engineering society «КАНАК». – № 3(25). – P. 45–48. [Электронный ресурс]. – Режим доступа: www.ntokaxak.kz, своб.
8. Advanced Digital Design with the Verilog HDL / M.D. Ciletti // Prentice Hall. – 2004.

- Чураев Сергей Олегович** – Санкт-Петербургский государственный университет информационных технологий, механики и оптики, аспирант, serg_vhdl@yahoo.com
- Адамова Айгуль Дюсенбиновна** – Евразийский национальный университет им. Л.Н. Гумилева, кандидат технических наук, докторант, aika_pavl@mail.ru
- Палташев Тимур Турсунович** – Санкт-Петербургский государственный университет информационных технологий, механики и оптики, Северо-Западный политехнический университет, Фримонт, Калифорния, доктор технических наук, профессор, timpal@mail.npu.edu