# ОСНОВЫ ОБРАБОТКИ ЗВУКА ВО ВСТРАИВАЕМЫХ СИСТЕМАХ. Часть 2*

**ДЭВИД КАТЦ, РИК ДЖЕНТАЙЛ, ТОМАШ ЛУКАСЕК (DAVID KATZ, RICK GENTILE, TOMASZ LUKASIAK)**, инженеры компании Analog Devices

***В статье рассматривается алгоритм цифровой обработки аудиосигналов в процессорах, используемых во встраиваемых системах. В первой части статьи был представлен краткий обзор аудиопреобразователей и рассматривались наиболее распространенные способы цифрового представления аудиосигналов. Во второй части рассказывается о способах представления данных в процессоре и методах обработки звука.***

## ФОРМАТЫ ДАННЫХ

Существует много способов представления данных внутри процессора. Как правило для обработки звука используются две архитектуры процессоров — с фиксированной и с плавающей точками. Процессоры с фиксированной точкой поддерживают целочисленную и дробную арифметику и обычно работают с 16-, 24- или 32-разрядными данными. Процессоры с плавающей точкой имеют высокую производительность и поддерживают 32- или 64-разрядные данные с плавающей точкой, однако обычно они потребляют больше мощности и стоят дороже. Поэтому в большинстве случаев разработчику приходится искать компромисс между качеством звука и стоимостью затрат.

### Арифметика с фиксированной точкой

В процессорах, поддерживающих арифметику с фиксированной точкой, для записи чисел обычно используется дополнительный формат. Этот формат позволяет представлять как знаковые, так и беззнаковые целые и дробные числа. Для целых чисел точка находится справа от младшего значащего разряда, а для дробных — обычно слева от знакового разряда. Представление с фиксированной точкой упрощает арифметические операции и сокращает необходимый объем памяти, однако динамический диапазон при этом будет ограничен. При необходимости обработки большого количества чисел с сохранением высокого разрешения желательно, чтобы положение точки изменялось в зависимости от значения и степени числа (например, как в формате с плавающей точкой).

### Арифметика с плавающей точкой

Запись чисел в формате с плавающей точкой очень похожа на научное представление рациональных чисел. Числа записываются при помощи мантиссы и экспоненты, которые определяют, соответственно, точность и динамический диапазон.

Операции с плавающей точкой в цифровых процессорах регулируются стандартом IEEE 754, в котором используется следующий формат записи чисел. Бит 31 (старший бит) — это знаковый бит (sign_bit), равный нулю для положительных чисел и единице для отрицательных. Биты 30—23 — это поле экспоненты (exp_field), соответствующее значению степени двойки со смещением, равным 127. И наконец, биты 22—0 — это дробная мантисса (mantissa). Формат также подразумевает наличие скрытого бита со значением 1, который находится слева от точки:

$$(-1)^{sign_bit}\ (1,mantissa)\ 2^{exp_field\ -\ 127}.$$

В таблице 1 перечислены минимальные и максимальные положительные величины, которые могут быть выражены при помощи некоторых стандартных типов данных.

### Эмуляция форматов на 16-разрядных процессорах

Хотя 16-разрядная арифметика не обеспечивает достаточного отношения сигнал/шум (ОСШ) для высококачественной обработки и воспроизведения звука, это не значит, что нужно отказаться от выбора 16-разрядных процессоров. Операции над 32-разрядными числами с плавающей точкой могут быть реализованы и на 16-разрядных процессорах, которые гораздо дешевле процессоров с аппаратной поддержкой арифметики с плавающей точкой путем эмуляции.

### Эмуляция плавающей точки на процессорах с фиксированной точкой

Для большинства 16-разрядных процессоров с фиксированной точкой существуют библиотеки, которые, в соответствии со стандартом IEEE 754, реализуют операции над числами с плавающей точкой путем их эмуляции при помощи арифметики с фиксированной точкой. Такая эмуляция требует дополнительных процессорных циклов, однако в связи с тем, что частота ядер современных процессоров с фиксированной точкой достигла диапазона 500 МГц…1 ГГц, во многих случаях дополнительная нагрузка не столь существенна.

Чтобы ускорить вычисления, можно использовать «облегченную» версию стандарта IEEE 754, в которой отсутствует, например, поддержка значений ±∞, «ноль» и NaN. Дальнейшая оптимизация заключается в представлении мантиссы и экспоненты с помощью более типичных для процессора типов данных. Возьмем, к примеру, процессоры с фиксированной точкой Blackfin производства Analog Devices, где имеется регистровый файл, состоящий из шестнадцати 16-разрядных регистров, которые можно также использовать как восемь 32-разрядных регистров. Для оптимального использования регистрового файла число с плавающей точкой может быть представлено парой 16-разрядных слов, из которых одно резервируется под экспоненту, а другое — под дробную часть числа.

Таблица 1. **Диапазон возможных значений при использовании различных типов данных**

| Тип данных | Наименьшее положительное значение | Наибольшее положительное значение |
|---|---|---|
| IEEE-754 с плавающей точкой (одинарная точность) | $2^{-126} \approx 1{,}2 \cdot 10^{-38}$ | $2^{128} \approx 3{,}4 \cdot 10^{38}$ |
| п. 1.15: 16-разрядные с фиксированной точкой | $2^{-15} \approx 3{,}1 \cdot 10^{-5}$ | $1 - 2^{-15} \approx 9{,}9 \cdot 10^{-1}$ |
| п. 1.23: 24-разрядные с фиксированной точкой | $2^{-23} \approx 1{,}2 \cdot 10^{-7}$ | $1 - {}^{-23} \approx 9{,}9 \cdot 10^{-1}$ |

** Часть 1 опубликована в журнале «Электронные компоненты» № 1, 2008.*

# Сверхнизкое энергопотребление. Супер-классный дизайн.

Сотни инженеров из более чем 100 стран приняли участие в конкурсе проектов на базе MSP430 eZ компании TI. Познакомьтесь с победившими проектами на сайте **www.ti.com/designmsp430** и оцените, как они использовали сверхнизкое энергопотребление и высокую степень интеграции MSP430. Также обязательно посетите виртуальную лабораторию MSP430 VirtuaLab на сайте **www.ti.com/designmsp430**, где можно бесплатно экспериментировать с микроконтроллером с самым низким энергопотреблением в мире и разрабатывать ваши собственные проекты. **И не забудьте оценить на нашу новую отладочную плату MSP430T2012, совместимую с eZ430.**

### T2012: Удобна в применении

- Совместима с eZ430-F2013
- Совместима по выводам со всеми устройствами F20xx
- 10-битовый АЦП, производительность 200 ksps (киловыборок в секунду)
- Прекрасный способ начать использовать MSP430

### T2012: Удобная процедура заказа

- Три отладочные платы T2012 всего за $10*
- www.ti.com/t2012-e
- Проектирование с использованием микроконтроллерного устройства, имеющего самое низкое энергопотребление, стало еще проще!

| Наименование | В составе комплекта | Поддерживаются устройства | Цена* |
|---|---|---|---|
| eZ430-F2013 | Эмулятор USB и отладочная плата | MSP430F20xx | $20 |
| Новинка! eZ430-T2012 | Отладочная плата eZ430 | MSP430F2012 | $10 / 3 в комплекте |

*В ценах надлежит учитывать курс обмена валюты и НДС.

Название Technology for Innovators и красно-черный баннер являются товарными знаками компании Texas Instruments.

© 2006 TI

# ПЯТЫЙ РОССИЙСКИЙ ФОРУМ

## ВСТРАИВАЕМЫЕ КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ: МНОГОЯДЕРНАЯ ЭВОЛЮЦИЯ

## 18 апреля 2008г., Москва

ФОРУМ ВКТ

ВСТРАИВАЕМЫЕ КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ

МОСКВА 2008

**Владейте информацией!**

Организатор: RTSoft

При участии: kontron, intel, LYNUXWORKS™, ELMA

**Ежегодный Форум «Встраиваемые компьютерные технологии»** - это традиционная встреча ведущих представителей технических и бизнес-кругов в области разработки отечественных аппаратно-программных комплексов мирового уровня.

**Приглашаются** руководители и специалисты компаний-производителей электронного оборудования и системных интеграторов, занятых в различных отраслях промышленности и разрабатывающих свои решения для авиации и космоса, обороны, специальных применений, промышленной автоматизации, телекоммуникаций, медицины, игровой индустрии и других направлений.

**Тематика Форума:**

- Многоядерная эволюция в мире Встраиваемых Компьютерных Технологий: современные тенденции;
- Новейшие аппаратные средства для разработчиков: от модулей и плат до базовых систем;
- Современный рынок встраиваемого программного обеспечения реального времени;
- Области применения оборудования и ПО: опыт отечественных системных интеграторов и OEM-производителей.

В рамках Форума проводится **выставка** ведущих мировых производителей встраиваемых компьютерных технологий и отечественных разработок на их основе.

**Для участия в Форуме** необходимо заполнить регистрационную форму на сайте: www.rtsoft.ru/ru/fvkt-08/?d или подать заявку
по телефонам: (495) 742-6828, 967-1505
по факсу: (495) 742-6829
по e-mail: fvkt-08@rtsoft.ru

**Информационная поддержка:**

Электроника, МКА, МИР ВКТ, PCWEEK, Control Engineering, Автоматизация промышленности, Информатизация и системы управления, Промышленные Контроллеры АСУ, Современная электроника, Компоненты и технологии, CHIP NEWS

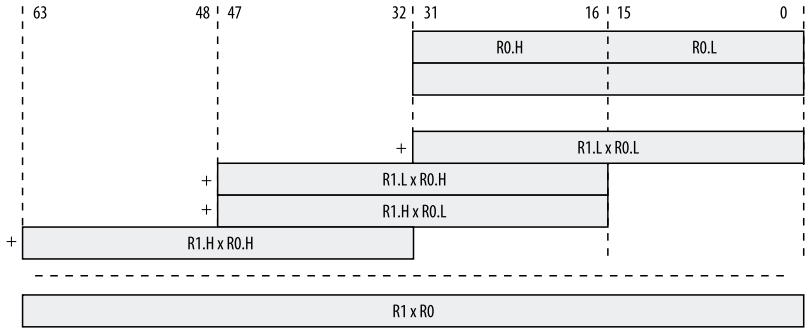


Рис. 1. **Умножение 32-разрядных чисел с фиксированной точкой, содержащихся в 32-разрядных регистрах R0 и R1**

**Эмуляция чисел с фиксированной точкой с двойной точностью**

Существует множество приложений, в которых недостаточно точности, получаемой при использовании 16-разрядных данных с фиксированной точкой, а вычислительные затраты на эмуляцию арифметики с плавающей точкой слишком велики. В таких случаях можно прибегнуть к эмуляции чисел с фиксированной точкой с повышенной точностью, позволяющей значительно повысить производительность вычислений, сохраняя достаточно большой динамический диапазон.

***Точная 32-разрядная эмуляция***

В 16-разрядных процессорах к регистрам регистрового файла можно обращаться как к двум 16-разрядным половинам одного регистра, и эти половины могут объединяться для представления единого 32-разрядного числа. Архитектура процессора Blackfin на аппаратном уровне поддерживает сложение и вычитание 32-разрядных чисел с фиксированной точкой, представленных подобным образом за один такт. Процесс умножения 32-разрядных чисел с фиксированной точкой, содержащихся в 32-разрядных регистрах R0 и R1, иллюстрирует рисунок 1. После необходимого масштабирования для представления в формате 1.31 окончательное выражение для 32-разрядного умножения будет иметь вид: ((R1.L x R0.L) >> 32 + (R1.L x R0.H) >> 16) + ((R1.H x R0.L) >> 16 + R1.H x R0.H). Возможность параллельного выполнения команд в архитектуре процессоров Blackfin позволяет реализовать подобную операцию за три цикла.

***Эмуляция с 31-разрядной точностью***

При понижении точности представления до 31 бита количество циклов, необходимых для перемножения чисел с фиксированной точкой, можно снизить до двух. Этот метод особенно подходит для аудиосистем, в которых обычно требуется представление чисел с разрядностью не менее 24 бит, а 32-разрядная точность может быть излишней.

Из рисунка 1 следует, что умножение младших половин слов R1.L × R0.L не вносит больших корректив в окончательный результат. Поскольку результат усекается для представления в формате 1.31, операция умножения повлияет только на младший бит результата. Во многих приложениях потеря точности, вызванная потерей одного разряда, компенсируется ускорением вычислений. Выражение для умножения с 31-разрядной точностью имеет вид: ((R1.L × R0.H) + (R1.H × R0.L)) >> 16 + + (R1.H × R0.H). В архитектуре процессоров Blackfin подобная операция может быть реализована за два цикла.

## МЕТОДЫ ОБРАБОТКИ АУДИОСИГНАЛОВ

Существуют разные способы ввода данных в ядро процессора. Например, основная программа может непосредственно опрашивать регистры данных порта, однако, поскольку такие пересылки приводят к неэффективному использованию ядра, они применяются редко. Вместо этого процессор, к которому подключен аудиокодек, может использовать механизм DMA для пересылки данных из внешнего интерфейса кодека в память процессора. Подобные операции происходят в фоновом режиме без вмешательства ядра процессора. Системные ресурсы участвуют при этом только в настройке DMA и обработке прерываний по завершении пересылки буфера данных.

**Блочная обработка и обработка отдельных отсчетов**

Существуют два подхода к обработке цифровых аудиосигналов — обработка отдельных отсчетов и блочная обработка. В первом случае процессор обрабатывает данные по мере их поступления, и в каждом периоде дискретизации возникают непроизводительные издержки. Подобным образом реализуются многие фильтры, поскольку эффективная задержка при этом меньше, чем при блочной обработке.

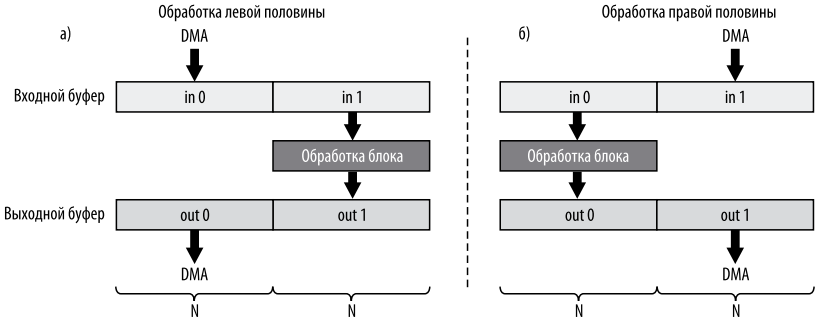


Рис. 2. **Схема с двойной буферизацией**

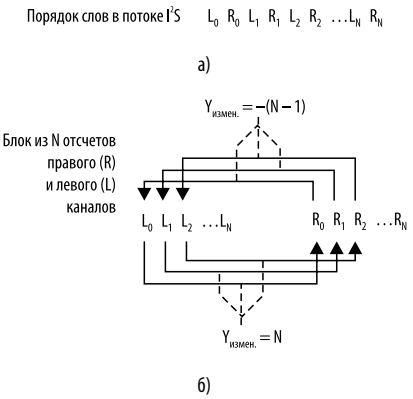


Рис. 3. **Схема механизма 2D DMA**

Во втором случае перед обработкой данных необходимо заполнить буфер определенной длины. При поступлении каждого отсчета не нужно вызывать подпрограмму обработки, поэтому непроизводительные издержки значительно сокращаются. Кроме того, многие процессоры для встраиваемых систем содержат несколько вычислительных узлов (например, умножителей или АЛУ), что позволяет обрабатывать блоки данных параллельно. И, что еще важнее, некоторые алгоритмы (например, преобразование Фурье) изначально ориентированы на блочную обработку данных.

### Двойная буферизация

Чтобы избежать одновременных обращений ядра и контроллера DMA к одному элементу в системах с блочной обработкой данных, необходимо использовать схему с двойной буферизацией. В этой схеме для работы с буфером длиной N создается сдвоенный буфер длиной 2N. Для двунаправленных систем необходимо создать два буфера длиной 2N. Пока ядро обрабатывает буфер in 1 и помещает результат в буфер out 1 (см. рис. 2а), контроллер DMA заполняет in 0 и передает данные из out 0. Как только контроллер DMA заканчивает работать с левыми половинами сдвоенных буферов, он начинает заполнять in 1 и передавать данные из out 1, а ядро переходит к обработке данных из in 0 и заполнению out 0 (см. рис. 2б). В системах реального времени обработка блока не должна занимать больше времени, чем затрачивает контроллер DMA на пересылку половины сдвоенного буфера.

### Двумерный DMA (2D DMA)

Поток, передаваемый по цифровому каналу, может содержать данные нескольких каналов, которые мультиплексируются в одну линию данных, как, например, в формате I²S. В этом случае для разбора данных отдельных каналов и их линейного упорядочивания в памяти может использоваться механизм 2D DMA (см. рис. 3).

## БАЗОВЫЕ ОПЕРАЦИИ

Обработка аудиосигналов основана на трех фундаментальных операциях: суммировании, умножении и задержки во времени. При помощи этих операций можно реализовать множество более сложных эффектов и алгоритмов. В большинстве мультимедийных процессоров операции сложения и умножения выполняются за один цикл. Задержка во времени несколько сложнее первых двух операций. Она реализуется при помощи линии задержки, которая представляет собой массив в памяти, содержащий данные от предыдущих отсчетов. Например, при создании эффекта эха для каждого канала может использоваться выборка входных отсчетов длительностью 500 мс. Текущий выходной отсчет вычисляется путем добавления к текущему входному отсчету слегка ослабленного (задержанного) предыдущего отсчета. Если в аудиосистеме применяется подход, основанный на обработке отдельных отсчетов, то программист может просто хранить указатели на входной и выходной отсчеты (разнесенные на число отсчетов, эквивалентные 500 мс) и инкрементировать их на каждом интервале дискретизации.

Поскольку линии задержки повторно используются для последующих наборов данных, входные и выходные указатели необходимо циклически перемещать по буферу, выполняя переход к началу буфера при достижении указателем его конца. Некоторые процессоры имеют аппаратную поддержку циклических буферов. В этом случае достаточно один раз задать начальную ячейку и длину буфера. В дальнейшем инкремент или декремент указателя в пределах буфера будет осуществляться программно, а возврат указателя к началу буфера при выходе за его пределы осуществляют аппаратные средства.

На линиях задержки основан такой важный составной элемент аудиосистем, как гребенчатый фильтр. Несколько гребенчатых фильтров, работающих одновременно, могут создавать эффект реверберации.

## ФОРМИРОВАНИЕ СИГНАЛОВ

В некоторых системах ставится задача синтеза сигналов (например, синусоидальных). Для формирования сигналов, описываемых тригонометрическими функциями, можно воспользоваться аппроксимацией рядами Тейлора, а для формирования белого шума используются генераторы случайных чисел с равномерным распределением. Если задача синтеза сигнала не вписывается в отведенный запас времени, вместо непосредственного вычисления сигнала можно использовать метод табличного поиска. Недостатком этого метода является расходование драгоценных ресурсов памяти, поэтому в качестве компромиссного решения можно использовать гибридные подходы. Можно, например, сократить объем памяти за счет хранения в ней только грубой таблицы. На этапе исполнения программы точные значения могут быть получены из табличных путем интерполяции, на которую затрачивается гораздо меньше времени, чем на аппроксимацию рядами Тейлора.

## ФИЛЬТРАЦИЯ И АЛГОРИТМЫ

Цифровые фильтры используются в аудиосистемах для ослабления или усиления энергии звукового сигнала на определенных частотах. Наиболее распространенными типами фильтров являются фильтры нижних частот, фильтры верхних частот, полосовые и режекторные фильтры. Любой из этих фильтров может быть реализован двумя способами: в виде фильтра с конечной

# Инновационные Аналоговые Компоненты

## Инновационные Аналоговые Компоненты от Лидера Поставок 8-разрядных Микроконтроллеров

Уменьшение энергопотребления и габаритов, а также снижения уровня помех и погрешностей обработки сигналов являются сегодня насущными требованиями к встраиваемым системам. Реагируя на требования рынка, компания Microchip предлагает интеграцию своих CMOS-технологий и опыта производства, реализованные в широкой гамме аналоговых и интерфейсных компонентов.

В состав предлагаемых компонентов входят: многочисленные компараторы и операционные усилители, в том числе и с программируемым коэффициентом усиления (PGA); микросхемы управления питанием; преобразователи; устройства измерения температуры; интерфейсные микросхемы и микросхемы смешанных сигналов.

Главными особенностями перечисленных компонентов являются малые энергопотребление и погрешность, низкий уровень помех, широкий диапазон рабочих напряжений и расширенный температурный диапазон, малые габариты и инновационные конструктивные особенности.

### Перечень основных видов продукции этого семейства:

- Температурные датчики
- Регуляторы напряжения
- Генераторы подкачки заряда и импульсные стабилизаторы
- Системные супервизоры
- Силовые драйверы КМОП
- Зарядные устройства
- Операционные усилители
- Усилители с программируемым коэффициентом усиления (PGA)
- Компараторы
- АЦП
- ЦАП
- Цифровые потенциометры
- CAN-интерфейсные продукты
- Драйверы инфракрасных излучателей и интерфейсные продукты

**Дополнительная информация доступна по ссылке www.microchip.com/analog**

microchip DIRECT
www.microchipdirect.com

Now Pb-free!
RoHS Compliant

MICROCHIP
www.microchip.com/analog

Наименование компании и логотипы являются зарегистрированными торговыми знаками компании. © 2007 Microchip Technology Inc. Все права защищены. ME160Rus/01.07

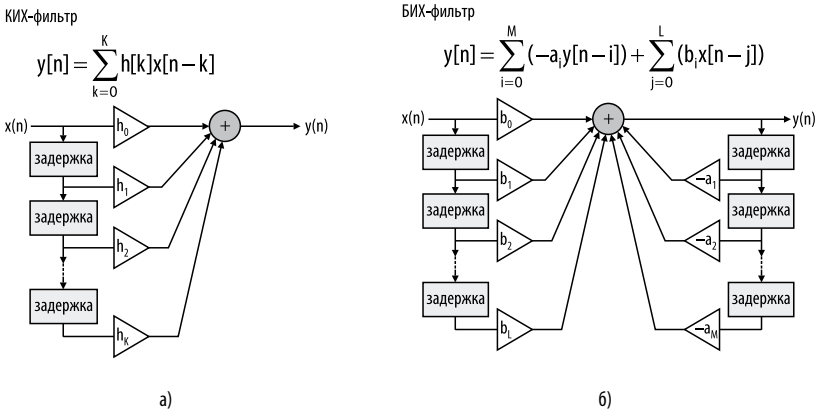


Рис. 4. **Структуры КИХ- и БИХ-фильтров**

импульсной характеристикой (КИХ) или фильтра с бесконечной импульсной характеристикой (БИХ).

**Фильтры с конечной импульсной характеристикой**

Выходной сигнал КИХ-фильтра определяется суммой текущего и предыдущих отсчетов сигнала, умноженных на соответствующие коэффициенты фильтра. Структура КИХ-фильтра и уравнение, описывающее его работу, которое также называется «сверткой», приведены на рисунке 4а. В этой записи x — входной вектор; y — выходной вектор; h — массив коэффициентов фильтра. Свертка — это настолько распространенная операция в обработке мультимедийных данных, что во многих процессорах реализовано выполнение команды умножения-накопления (MAC) одновременно с несколькими обращениями (чтением и записью) к памяти за один цикл.

**Фильтры с бесконечной импульсной характеристикой**

В отличие от КИХ-фильтров, выходной сигнал которых зависит только от входных отсчетов, выходной сигнал БИХ-фильтров зависит как от входных, так и от выходных отсчетов в предыдущие моменты времени, из-за чего их часто называют рекурсивными фильтрами. Структура КИХ-фильтра и разностное уравнение, описывающее его работу, приведены на рисунке 4б.

**Быстрое преобразование Фурье**

Зачастую с аудиосигналами удобнее работать в частотной области. Для преобразования сигнала из временной области в частотную используется преобразование Фурье. Обратное преобразование Фурье выполняет преобразование из частотной области во временную. Существует определенная взаимосвязь между операциями в частотной и во временной областях. Так, свертка во временной области (или пропускание сигнала через КИХ-фильтр) эквивалентна умножению в частотной области. Это свойство не имело бы большого практического значения, если бы не существовала оптимизированная реализация преобразования Фурье, которая называется быстрым преобразованием Фурье (БПФ). Во многих случаях реализовать КИХ-фильтр с большим количеством коэффициентов проще путем вычисления спектра сигнала, его умножения на частотную характеристику фильтра и взятия обратного преобразования от результата умножения.

При обработке аудиосигналов часто применяются и другие преобразования, например модифицированное дискретное косинусное преобразование (MDCT), которое является базисным для многих алгоритмов сжатия аудиосигналов.

**Преобразование частоты дискретизации**

В некоторых задачах возникает необходимость преобразовать сигнал, оцифрованный с одной частотой, в сигнал, оцифрованный с другой частотой. Это может потребоваться, например, в случаях, когда нужно выдать сигнал, оцифрованный с низкой частотой, на более высокочастотный ЦАП, или, если требуется, сократить количество вычислений за счет понижения частоты сигнала с избыточной дискретизацией.

Процесс повышения частоты дискретизации называется интерполяцией, а процесс ее понижения — децимацией. Децимация сигнала в M раз достигается взятием каждого M-го отсчета и отбрасыванием остальных. Интерполяция сигнала в L раз обеспечивается добавлением в сигнал L – 1 нулей между каждыми двумя отсчетами. Чтобы реализовать рациональные коэффициенты преобразования, операции интерполяции и децимации могут применяться последовательно.

Нужно отметить, что процесс преобразования частоты дискретизации описан здесь упрощенно. На практике перед выдачей интерполированного сигнала или перед его подачей на вход дециматора он должен быть пропущен через ФНЧ с особой структурой для устранения гармоник, появляющихся из-за добавления нулей. Аналогичным образом перед децимацией входные сигналы для предотвращения спектральных наложений должны быть отфильтрованы в ФНЧ специальной формы.