

ВИКИПЕДИЯ

Цифровой сигнальный процессор

Материал из Википедии — свободной энциклопедии

Цифровой сигнальный процессор (англ. *digital signal processor*, *DSP*, цифровой процессор обработки сигналов (ЦПОС) — специализированный микропроцессор, предназначенный для обработки оцифрованных сигналов (обычно, в режиме реального времени)^[1].

Содержание

Особенности архитектуры

Области применения

История

Предшествующие разработки

Первое поколение (начало 1980-х)

Второе поколение (середина 1980-х)

Третье поколение (конец 1980-х)

Четвёртое поколение

Современные ЦСП

Основные параметры ЦСП

Оценка и сравнение производительности

Устройство

Гарвардская архитектура

Структурная схема

Конвейерное исполнение команд

Аппаратная реализация основных функций

Умножители



ЦСП обработки видео из картриджа приставки Nintendo

[Сдвигатели](#)[Устройства генерации адреса](#)[Аппаратная организация циклов](#)[АЛУ](#)[Регистры](#)[Способы адресации](#)

Классификация ЦСП по архитектуре

[Стандартные ЦСП](#)[Улучшенные стандартные ЦСП](#)

ЦСП с архитектурой VLIW

Суперскалярные ЦСП

Гибридные ЦСП

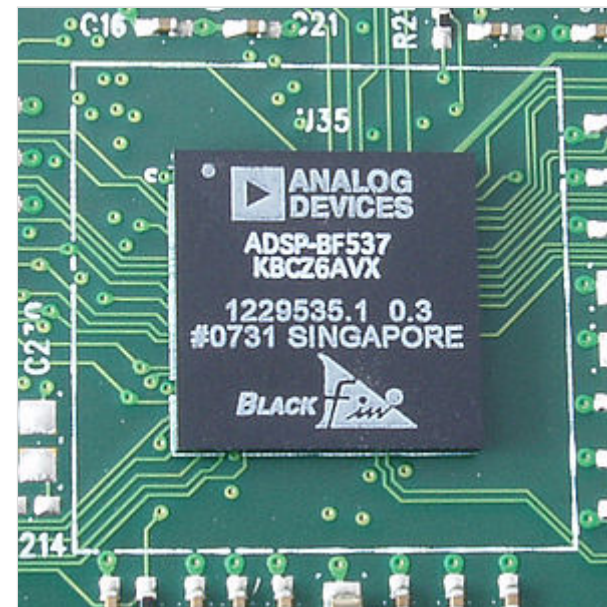
Классификация ЦСП по назначению

Программирование ЦСП

[Особенности ассемблеров ЦСП](#)[Совместимость внутри семейств ЦСП](#)[Отладка программ](#)

Примечания

Литература



Современный ЦСП с BGA-выводами, установленный на поверхность печатной платы

Особенности архитектуры

Архитектура сигнальных процессоров, по сравнению с микропроцессорами общего применения, имеет некоторые особенности, связанные со стремлением максимально ускорить выполнение типовых задач цифровой обработки сигналов, таких, как цифровая фильтрация, преобразование Фурье, поиск сигналов и т. п. Математически эти задачи сводятся к поэлементному перемножению элементов многокомпонентных векторов действительных чисел, последующему суммированию произведений (например, в цифровой фильтрации выходной сигнал фильтра с конечной импульсной характеристикой равен сумме произведений коэффициентов фильтра на вектор выборок сигнала, аналогичные вычисления производятся при поиске максимумов корреляционных и автокорреляционных функций выборок сигналов). Поэтому сигнальные процессоры оптимизированы по быстродействию для выполнения именно таких операций. И ЦСП ориентированы, в первую очередь, на многократное выполнение умножения с расчётом «на лету» адресов перемножаемых элементов массивов:

- Операция «умножение с накоплением» (англ. *multiply-accumulate*, *MAC*) ($Y = Y + A \times B$), где Y , A , B — элементы действительных массивов с автоматическим расчётом адресов элементов массивов и обычно реализована аппаратно и выполняется за один машинный цикл.
- Аппаратная реализация многократного повторения заданного набора команд, то есть циклы с заранее назначенной длиной без использования счётчиков цикла и команд проверки обнуления счётчика цикла — признака выхода из цикла.
- Возможность одновременной в одном машинном такте выборки команды и двух операндов для максимально быстрого выполнения команды MAC. Для этого ЦСП имеет несколько портов обращения к памяти (независимых областей памяти, каждая со своим комплектом шин адреса и данных).
- Поддержка векторно-конвейерной обработки с помощью генераторов адресных последовательностей.

Ограниченность аппаратных ресурсов первых ЦСП накладывала существенный отпечаток на их архитектуру:

- Гарвардская архитектура (разделение памяти команд и данных), как правило, модифицированная; с разделением памяти на сегменты с независимым доступом.
- Детерминированная работа с известными временами выполнения команд, что позволяет выполнять планирование работы в реальном времени.
- Сравнительно небольшая длина конвейера, так что незапланированные условные переходы могут занимать меньше время, чем в универсальных процессорах.
- Экзотический набор регистров и инструкций, часто сложных для компиляторов. Некоторые архитектуры тогда использовали VLIW.

В современной микроэлектронике процессоры общего применения зачастую содержат аппаратную поддержку типовых операций ЦОС. Особо времяёмкие задачи ЦОС решаются на основе программируемой логики, где можно достичь предельной оптимизации выполнения конкретной операции. Специализированные процессоры ЦОС всё чаще делают векторными. В то же время классические ЦСП снабжают развитыми наборами команд процессоров общего применения и сглаживают особенности программной модели, позиционируя их как изделия широкого применения с ускоренными функциями ЦОС. Все эти тенденции приводят к размыванию классического понятия ЦСП.

Области применения

- Коммуникационное оборудование:
 - Уплотнение каналов передачи данных;
 - Кодирование аудио- и видеопотоков;
- Системы гидро- и радиолокации;
- Распознавание речи и изображений;
- Речевые и музыкальные синтезаторы;

- Анализаторы спектра;
- Управление технологическими процессами;
- Другие области, где необходима быстродействующая обработка сигналов, в том числе в реальном времени.

История

Предшествующие разработки

До 1980 года несколько компаний выпустили устройства, которые можно считать предшественниками ЦСП. Так, в 1978 Intel выпускает «процессор аналоговых сигналов» 2120. В его состав входили АЦП, ЦАП и процессор обработки цифровых данных, однако аппаратная функция умножения отсутствовала. В 1979 AMI выпускает S2811 — периферийное устройство, управляемое основным процессором компьютера. Оба изделия не достигли успеха на рынке.

Первое поколение (начало 1980-х)

Основную историю ЦСП принято отсчитывать от 1979—1980 годов, когда Bell Labs представила первый однокристалльный ЦСП Mac 4, а также на «IEEE International Solid-State Circuits Conference '80» были показаны μMPD7720 компании NEC и DSP1 компании AT&T, которые, однако, не получили широкого распространения. Стандартом де-факто стал выпущенный чуть позже кристалл TMS32010 фирмы Texas Instruments, по многим параметрам и удачным техническим решениям превосходящий изделия конкурентов. Вот некоторые его характеристики:

- АЛУ:
 - Размер слова: 16 бит;
 - Разрядность вычислителя: 32 бит;
 - Быстродействие: 5 млн операций сложения или умножения в секунду;
- Длительность командного цикла: 160—280 нс;
- Память:
 - ОЗУ: 144—256 слов;
 - ПЗУ программ: 1,5—4 К слов;
 - ППЗУ: до 4К слов (отдельные модели);
- Внешняя шина:

- Разрядность: 16 бит;
- Адресуемое пространство: 4К слов
- Пропускная способность: 50 Мбит/с
- Устройства ввода-вывода: 8 портов по 16 разрядов;

Второе поколение (середина 1980-х)

Благодаря прогрессу в полупроводниковых технологиях в этот период были выпущены изделия, имеющие расширенные функции по сравнению с первым поколением. К характерным отличиям можно отнести:

- Увеличение объёма ОЗУ до 0,5 К слов;
- Добавлена возможность подключения внешней памяти программ и внешней памяти данных объёмом до 128 К слов;
- Быстродействие повышено в 2—4 раза;
- Улучшенные подсистемы прерываний и ввода-вывода.

Много позднее также были выпущены устройства, формально относящиеся ко второму поколению, но имеющие следующие усовершенствования:

- Увеличена разрядность данных;
- Пониженное напряжение питания и, как следствие, энергопотребление;
- Введены режимы экономии энергии;
- Аппаратная поддержка мультипроцессорности (система совместного доступа к внешней памяти);
- Аппаратная поддержка кольцевых буферов;
- Аппаратная поддержка операций циклов;
- Расширены способы адресации;
- Две внутренние шины данных, что позволяет значительно ускорить парную обработку данных (координаты X/Y, действительная и мнимая часть и т. д.), либо виртуально удвоить разрядность обрабатываемых данных;
- Введена кэш-память.

Третье поколение (конец 1980-х)

Третье поколение ЦСП принято связывать с началом выпуска изделий, реализующих арифметику с плавающей запятой. Характерные особенности первых выпущенных образцов:

- Производительность: порядка 20-40 млн оп./сек. (MIPS);
- Два блока ОЗУ по 1 К 32-разрядных слов с возможностью одновременного доступа;
- Кэш-память объёмом 64 слова;
- Разрядность регистров: 32 бит;
- Разрядность АЛУ: 40 бит;
- Регистры для операций с повышенной точностью;
- Встроенные контроллеры ПДП;
- Разрядность шин: 32 бит для команд и 24 бит для адреса;

Четвёртое поколение

Четвёртое поколение ЦСП характеризуется значительным расширением наборов команд, созданием VLIW и суперскалярных процессоров. Заметно возросли тактовые частоты. Так, например, время выполнения команды MAC ($Y := X + A \times B$) удалось сократить до 3 нс.

Современные ЦСП

Лучшие современные ЦСП можно характеризовать следующими параметрами:

- Тактовая частота — 1 ГГц и выше;
- Многоядерность;
- Наличие двухуровневого кеша;
- Встроенные многоканальные контроллеры прямого доступа к памяти;
- Быстродействие порядка нескольких тысяч MIPS и MFLOPS;
- Выполнение до 8 параллельных инструкций за такт;
- Совместимость со стандартными шинами (PCI и др.)

Основные параметры ЦСП

- **Тип арифметики.** ЦСП делятся на процессоры, обрабатывающие данные с фиксированной точкой и обрабатывающие данные с плавающей точкой. Устройства с плавающей точкой удобнее в применении, но они заметно сложнее по устройству и более дороги;
- **Разрядность данных.** Большинство ЦСП с фиксированной точкой обрабатывают данные с разрядностью 16 бит, процессоры с плавающей точкой — 32 бита. Многие модели могут обрабатывать данные с двойной точностью.

- **Быстродействие.** Быстродействие как интегральную характеристику определить достаточно сложно, поэтому скорость работы характеризуют несколькими параметрами, а также временем решения некоторых реальных задач.
 - **Тактовая частота и время командного цикла.** Для современных ЦСП тактовая внутренняя частота может отличаться от внешней, поэтому могут указываться два значения. Время командного цикла указывает на время выполнения одного этапа команды, то есть время одного цикла конвейера команд. Так как команды могут исполняться за разное количество циклов, а также с учётом возможности одновременного исполнения нескольких команд, этот параметр может характеризовать быстродействие ЦСП достаточно приближённо.
 - **Количество выполняемых команд за единицу времени.** Различное время исполнения команд, а также исполнение нескольких команд одновременно не позволяют использовать этот параметр для надёжной характеристики быстродействия.
 - **Количество выполняемых операций за единицу времени (MIPS).** Данный параметр учитывает одновременную обработку нескольких команд и наличие параллельных вычислительных модулей, поэтому достаточно хорошо может указывать на быстродействие ЦСП. Некоторой проблемой здесь остаётся то, что понятие «операции» чётко не формализовано.
 - **Количество выполняемых операций с плавающей точкой за единицу времени.** Параметр аналогичен предыдущему и используется для процессоров с плавающей точкой.
 - **Количество выполняемых операций MAC за единицу времени.** Данная команда, с одной стороны, является базовой для многих вычислений, а с другой — достаточно проста. Поэтому время её исполнения можно использовать в том числе и для оценки общей производительности ЦСП.
- **Виды и объём внутренней памяти.** Объём внутренней оперативной памяти показывает, сколько данных ЦСП может обработать без обращения к внешней памяти, что может характеризовать общее быстродействие системы, а также возможность работать «в реальном времени». Тип ПЗУ определяет возможности по программированию устройства. Модели с обычным ПЗУ подходят для крупносерийного производства, ППЗУ (однократно программируемое) удобно для небольших тиражей, а применение Flash-памяти позволяет менять программу устройства многократно во время эксплуатации. На данный момент не очень мощные DSP чаще всего снабжены достаточно большой Flash-памятью (её цена неуклонно снижается) и заметным объёмом RAM и поэтому могут являться самодостаточными без добавления внешней памяти и на этапе разработки, и на этапе производства, что повышает конкурентоспособность таких DSP в ряде сегментов рынка. Мощные DSP, как правило, полагаются на внешнюю память, подключённую по достаточно быстрым шинам, а размещение там Flash-памяти может быть технически проблематичным, например, из-за заметного выделения тепла.
- **Адресуемый объём памяти.** Объём адресуемой внешней памяти характеризуется шириной внешней шины адреса.
- **Способ начальной загрузки.**
- **Количество и параметры портов ввода-вывода.** Данный параметр показывает возможности ЦСП по взаимодействию с внешними по отношению к нему устройствами.
- **Состав внутренних дополнительных устройств.** В число внутренних могут входить разнообразные по назначению устройства, например, *общего применения* — таймеры, контроллеры ПДП и т. д., а также *проблемно-ориентированные* — АЦП, кодеки, компрессоры данных и другие.
- **Напряжение питания и потребляемая мощность.** Данная характеристика особенно важна для ЦСП, встраиваемых в переносные устройства. Обычно предпочтительнее низковольтные устройства (1,8-3,3В), которые имеют быстродействие аналогично 5В

процессорам, но заметно экономнее в плане потребления энергии. Многие устройства имеют режимы экономии при простое, либо позволяют программно отключать часть своих устройств.

■ Состав и функциональность средств разработки и поддержки.

- Перечень языков программирования, для которых есть компиляторы под данную систему;
- Наличие и возможности средств отладки готовых программ;
- Доступность документации и технической поддержки;
- Наличие библиотек стандартных подпрограмм и математических функций;
- Наличие, доступность и возможности совместимых устройств — АЦП, ЦАП, контроллеры питания и т. д.

■ Допустимые параметры окружающей среды.

- Другие, в зависимости от назначения.

Часто используются также интегральные характеристики ЦСП, например, показатель «мощность/ток/быстродействие», например, mA/MIPS (миллиампер на 1 млн инструкций в секунду), что позволяет оценить реальную потребляемую мощность в зависимости от сложности задачи, решаемой процессором в указанный момент.

Выбор ЦСП целиком определяется назначением разрабатываемой системы. Например, для массовых мобильных устройств важна дешевизна процессора, низкое энергопотребление, в то время как стоимость разработки системы отходит на второй план. С другой стороны, для измерительного оборудования, систем обработки звуковой и видеоинформации важны эффективность процессора, наличие развитых инструментальных средств, многопроцессорность и т. д.

Оценка и сравнение производительности

Как отмечено ранее, отдельные характеристики типа тактовой частоты, MIPS, MOPS, MFLOPS позволяют оценить быстродействие ЦСП достаточно неоднозначно. Поэтому для решения задачи измерения и сравнения характеристик разных ЦСП используют специальные наборы тестов, имитирующих некоторые распространённые задачи цифровой обработки сигналов. Каждый тест состоит из нескольких небольших программ, которые пишутся на ассемблере и оптимизируются под заданную архитектуру. Эти тесты могут включать реализацию:

- Фильтры КИХ и БИХ;
- Перемножение векторов;
- Декодеры Витерби;
- БПФ

Наиболее авторитетным пакетом тестов на сегодняшний день является тест **BDTImark2000** (**BDTI DSP Kernel Benchmarks™** (**BDTImark2000™**) **Certified Results** (<http://www.bdti.com/Resources/BenchmarkResults>)), который, кроме указанных алгоритмов, включает также оценку используемой алгоритмом памяти, время разработки системы и другие параметры.

Устройство

Гарвардская архитектура

Цифровые сигнальные процессоры строятся на основе т. н. «Гарвардской архитектуры», отличительной особенностью которой является то, что программы и данные хранятся в различных устройствах памяти — памяти программ и памяти данных. В отличие от архитектуры фон Неймана, где процессору для выборки команды и двух операндов требуется минимум три цикла шины, ЦСП может производить одновременные обращения как к памяти команд, так и к памяти данных, и указанная выше команда может быть получена за два цикла шины. В реальности, благодаря продуманности системы команд и другим мерам, это время может быть сокращено до одного цикла. В реальных устройствах память команд может хранить не только программы, но и данные. В этом случае говорят, что ЦСП построен по модифицированной гарвардской архитектуре.

Память команд и память данных обычно располагаются на кристалле ЦСП. В связи с тем, что эта память имеет относительно небольшой объём, возникает необходимость в использовании внешних (относительно кристалла процессора) запоминающих устройств. Для таких устройств отдельные шины команд и данных не используются, так как это потребовало бы значительно увеличить количество внешних выводов кристалла, что дорого и непрактично. Поэтому взаимодействие ЦСП с внешними запоминающими устройствами происходит по одному комплекту шин без разделения на команды и данные. Следует также заметить, что обращение к внешней памяти всегда занимает значительно больше времени, чем к внутренней, поэтому в приложениях, критичных ко времени исполнения, такие обращения необходимо минимизировать.

Структурная схема

Конвейерное исполнение команд



ЦСП строятся на основе Гарвардской архитектуры

Конвейер представляет собой вычислительный поток, который на каждой стадии выполняет определённую **микрооперацию**, поэтому на конвейере в каждый момент времени находится несколько команд на разной стадии выполнения. Это позволяет повысить быстродействие.

Наличие нескольких конвейеров реализует **суперскалярную архитектуру**.

При параллельной обработке команд на разных конвейерах максимальный эффект достигается на однотипных командах, не зависящих друг от друга. Если в программе присутствуют команды разного типа, то на конвейере вводятся **такты ожидания**.

Для оптимизации загрузки конвейеров необходимо следующее:

- Компиляция машинного кода под конкретный процессор.
- Оптимизация загрузки конвейера в блоки предварительного декодирования команд.

В итоге команды выполняются не в том порядке, в каком их записывал программист.

Аппаратная реализация основных функций

Умножители

Аппаратный умножитель применяется для сокращения времени выполнения одной из основных операций ЦОС — операции умножения. В процессорах общего назначения эта операция реализуется за несколько тактов сдвига и сложения и занимает много времени, а в DSP благодаря специализированному умножителю — за один командный цикл.

Функционально умножители делятся на два вида:

- Простой умножитель. Выполняет операцию умножения данных шириной в слово. Результат имеет ширину двойного слова и сохраняется либо в регистре двойной ширины, либо в двух обычных регистрах (или в двух ячейках памяти).
- Умножитель-сумматор (MAC — Multiplier/Accumulator). Выполняет операцию умножения с накоплением, которая широко используется во многих алгоритмах цифровой обработки сигналов. Подробнее об использовании данной команды см. в разделе [#Классификация ЦСП по архитектуре](#).

Сдвигатели

Сдвигателем называется как устройство, выполняющее операцию сдвига данных, так и регистр, в котором хранится результат сдвига^[2].

С точки зрения выполняемых функций, сдвигатели делятся на:

- *Предсдвигатели*, выполняющие сдвиг до начала операции или в ходе её исполнения;
- *Постсдвигатели*, выполняющие сдвиг после исполнения операции.

В обоих случаях структура регистра, хранящего результат сдвига, совпадает со структурой аккумулятора.

Функции предсдвигателей

- Предварительное масштабирование. Используется в сложных арифметических командах, например, $A \leftarrow A \cdot 2 - B$, а также командах загрузки со сдвигом вида $A \leftarrow B \ll n$;
- Сдвиг перед выполнением сложных логических операций, например, $A \leftarrow (A \ll n) \wedge B$;
- Арифметические, логические и циклические сдвиги в ходе исполнения соответствующих команд.

Функции постсдвигателей

- Масштабирование результатов при сохранении в память. При этом содержимое аккумулятора (результат основной операции) остаётся неизменным;
- Удаление битов расширения знака;
- Нормализация;
- Выделение одинакового порядка.

Устройства генерации адреса

Аппаратная организация циклов

АЛУ

АЛУ — блок процессора, который под управлением *декодера команд* выполняет арифметические и логические преобразования над данными, называемыми в этом случае операндами. Разрядность операндов обычно называют размером машинного слова.

Регистры

Аккумулятор

Аккумулятор — регистр, предназначенный для сохранения результатов операций. В архитектуре многих ЦСП предусмотрено два аккумулятора, что позволяет повысить скорость выполнения операций, требующих хранения промежуточных результатов. Технически аккумулятор может состоять из нескольких регистров^[2]:

- EXT — регистр расширения;
- MSP — регистр старшего слова;
- LSP — регистр младшего слова.

Наличие регистра EXT позволяет повысить точность вычисления промежуточных результатов, а также увеличить диапазон хранения значений, не приводящих к переполнению. При сохранении значения аккумулятора в ячейку памяти или в обычный регистр, его значение округляется с учётом стандартной ширины этой ячейки или регистра. С другой стороны, при необходимости, содержимое регистра EXT может быть сохранено отдельно.

Способы адресации

Процессор поддерживает режимы прямой адресации, косвенной адресации с пред- и постинкрементом и специфические для задач цифровой обработки сигналов режимы циклической адресации и адресации с реверсированием бит адреса.

Классификация ЦСП по архитектуре

Следует отметить, что приведённая ниже классификация^[2] достаточно условна, так как разнообразие технических решений зачастую не позволяет однозначно отнести каждое конкретное устройство к одному из указанных типов. Поэтому нижесказанное следует скорее использовать как материал для понимания особенностей архитектуры ЦСП, чем для реальной классификации каких-либо изделий.

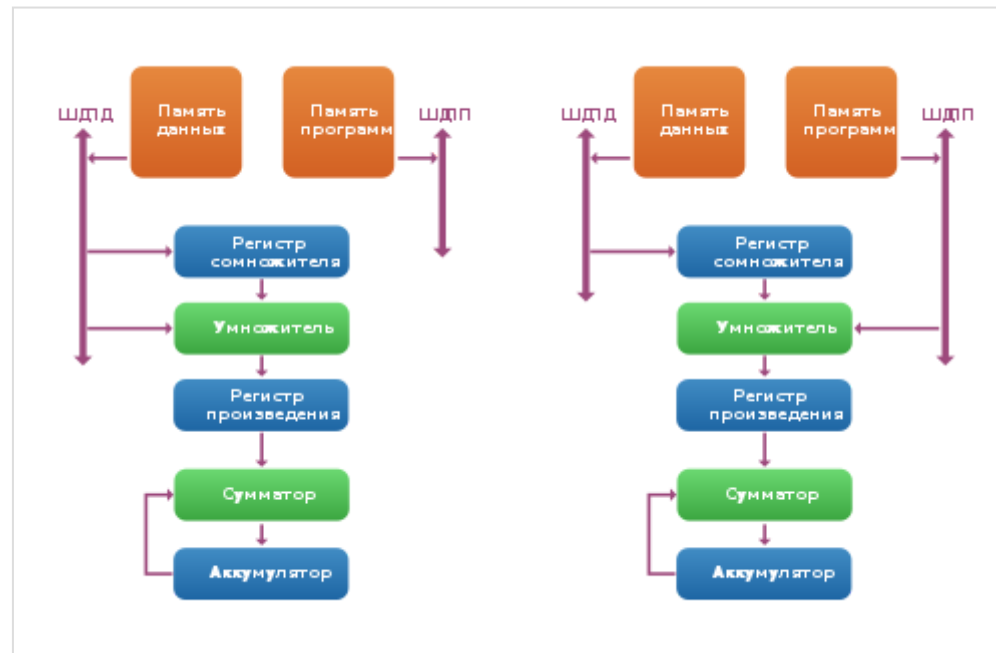
Особенности архитектуры ЦСП удобно рассматривать на примере конкретного алгоритма цифровой обработки данных, например, КИХ-фильтра, выходной сигнал которого можно записать как:

$$y(n) = \sum_{i=0}^P b_i x(n-i), \text{ где}$$

- $x(n-i)$ — отсчёты входного сигнала;
- b_i — коэффициенты фильтра.

Как можно легко заметить, вычисление результата является классическим примером использования операции «умножение с накоплением» — MAC ($Y := X + A \times B$).

Стандартные ЦСП



Два варианта исполнения команды MAC на ЦСП Texas Instruments

На рисунке показано два варианта выполнения команды MAC на стандартном ЦСП. В первом варианте оба операнда хранятся в памяти данных, поэтому на их выборку требуется два такта, то есть время выполнения n сложений равно $2n$. Во втором случае один из операндов хранится в памяти программ, поэтому команда исполняется за один такт, и общее время выполнения цикла будет равно n тактов (следует уточнить, что в реальности для исполнения за один такт MAC должна исполняться внутри специальной команды цикла для исключения повторной выборки самого кода команды, что требует дополнительного такта). Здесь видно, что эффективная реализация алгоритма требует использования памяти программ для хранения данных.

Одним из вариантов, позволяющим отказаться от использования памяти программ для хранения данных, является применение т. н. «двухпортовой памяти», то есть памяти, имеющей два комплекта входных шин — двух шин адреса и данных. Такая архитектура позволяет произвести одновременное обращение по двум адресам (правда, при этом они должны находиться в разных адресуемых блоках). Данное решение применяется в ЦСП компаний Motorola (DSP56000) и Lucent (DSP1600).

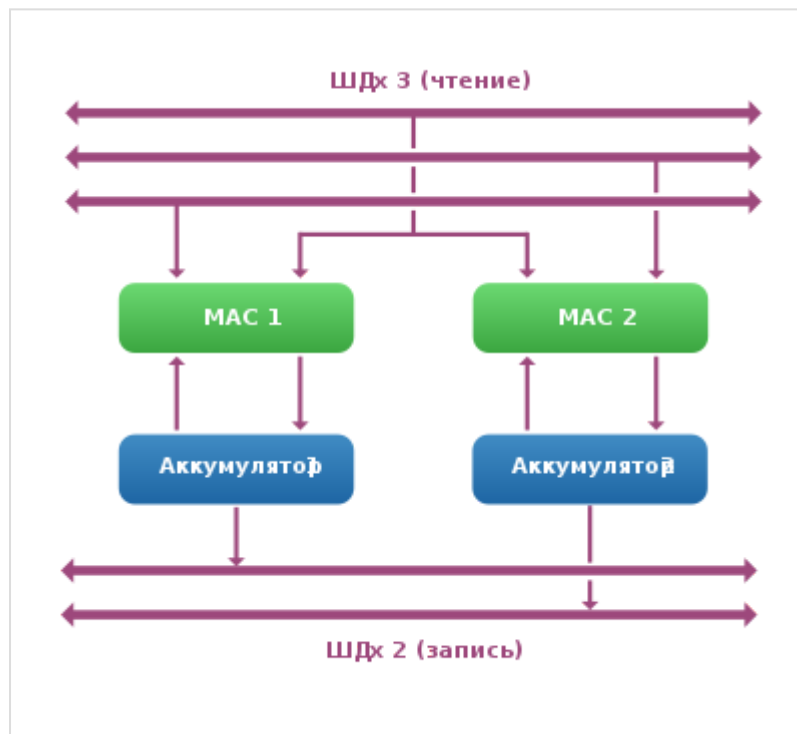
При указанной архитектуре повысить производительность можно только увеличением тактовой частоты.

Улучшенные стандартные ЦСП

«Улучшенные стандартные ЦСП» для повышения производительности системы, по сравнению со стандартными ЦСП, используют следующие методы повышения параллелизма:

- Увеличение количества операционных и вычислительных устройств;
- Введение специализированных сопроцессоров;
- Расширение шин для увеличения количества передаваемых данных;
- Использование памяти с многократным доступом (несколько обращений за такт);
- Усложнение системы команд;

Многие из этих способов применялись уже начиная с самых первых процессоров, поэтому зачастую их невозможно однозначно классифицировать как «стандартные» или «улучшенные».



Исполнение двух операций MAC на ЦСП Texas Instruments TMS320C55x

На рисунке показан пример реализации вычисления двух параллельных команд MAC. Для этого в ЦСП присутствуют два модуля MAC и два аккумулятора. Блоки MAC получают данные по трём шинам одновременно, причём одно из значений является для них общим. Таким образом, происходит одновременное исполнение двух команд:

- $AK1 := AK1 + D1 \times D2$
- $AK2 := AK2 + D1 \times D3$

Особенность показанного решения состоит в том, что к выполнению двух параллельных команд с одним общим сомножителем можно свести многие алгоритмы ЦОС, например:

- КИХ-фильтр с симметричными коэффициентами. В качестве одинаковых сомножителей используются коэффициенты фильтра, а на отдельные шины подаются два разных набора отсчётов сигнала, то есть параллельно рассчитываются две половины фильтра, которые затем суммируются.
- Двухканальная обработка. На общую шину подаются отсчёты сигнала, а на отдельные — наборы коэффициентов.

В некоторых процессорах (Lucent DSP16xxx, ADI ADSP-2116x) используются два одинаковых ядра, каждый со своей памятью, то есть одна команда исполняется одновременно в двух ядрах с различными данными. Это позволяет обойти ограничение на использование полностью независимых данных.

Характерным недостатком таких процессоров можно считать необходимость в высокой квалификации разработчика, так как эффективное использование указанных особенностей требует программирования на языке ассемблера, хорошего знания архитектуры и системы команд, то есть эти устройства считаются «недружественными» к языкам высокого уровня.

ЦСП с архитектурой VLIW

Основное отличие VLIW-процессоров состоит в том, что коды команд ещё на этапе компиляции собираются в большие «суперкоманды» и выполняются параллельно. Обычно такие процессоры используют RISC-архитектуру с фиксированной длиной команды, где каждая из них выполняется в отдельном операционном модуле. К характерным особенностям таких процессоров можно отнести:

- Большой набор операционных модулей, работающих независимо друг от друга. В состав таких модулей могут входить:
 - Арифметические:
 - модули арифметических операций и операций сравнения;
 - модули логических операций;
 - модули умножения чисел с плавающей и фиксированной точкой;
 - модули генерации констант.
 - Модули генерации адреса, в том числе для линейных и циклических буферов;
- Необходимость оптимизации компилятора под каждую модель процессора, так как между моделями может меняться состав и функции вычислительных блоков, что влечёт за собой изменение перечня команд, которые могут выполняться одновременно;
- Необходимость в наличии сверхшироких шин данных (порядка 128 бит), чтобы код операции, состоящий из отдельных команд (до 8-ми), мог быть получен из памяти за одно обращение.
- Высокие требования к объёму памяти программ, что также связано с большой длиной операции.

Обычно, если процессор имеет несколько одинаковых модулей, то при создании программы на ассемблере имеется возможность указания только типа необходимого операционного модуля, а конкретное устройство будет назначено компилятором. С одной стороны, это упрощает программирование таких устройств, а с другой стороны, позволяет достаточно эффективно использовать их ресурсы.

Суперскалярные ЦСП

Суперскалярные процессоры также характеризуются большим набором параллельных операционных модулей и возможностью одновременного исполнения нескольких команд. Однако, по сравнению с VLIW, они имеют две характерные особенности:

- Команды процессора не группируются в блоки, каждая из них поступает в процессор независимо;
- Команды для параллельного исполнения группируются внутри процессора на основе состава и текущей загруженности операционных блоков, а также зависимости между данными.

С помощью описанного подхода можно обойти следующие недостатки VLIW:

- Неэффективное использование памяти из-за большой длины групповой операции;
- Зависимость скомпилированного кода от состава операционных модулей конкретного процессора.

Платой за решение этих проблем становится значительное усложнение схемы процессора, в котором появляется модуль планирования выполнения команд.

Суперскалярные процессоры планируют исполнение команд не только на основе информации о загруженности операционных блоков, но и на основе анализа зависимостей между данными. К примеру, команда сохранения результата арифметической операции не может быть выполнена раньше самой операции вычисления, даже если модуль обращения к памяти в данный момент свободен. Эта особенность приводит в том числе к тому, что один и тот же набор команд может по-разному исполняться в различных местах программы, что делает невозможным точную оценку производительности. Особенно это важно для систем, работающих в реальном времени, ведь оценка по наихудшему результату приведёт к тому, что ресурсы процессора будут использованы не полностью. Таким образом, в этих системах задача точной оценки производительности суперскалярных ЦСП остаётся открытой.

Гибридные ЦСП

Под гибридными ЦСП обычно понимают специализированные устройства, сочетающие в себе функцию микроконтроллера и цифрового сигнального процессора. Обычно такие изделия предназначены для выполнения одной функции — например, управления электрическими двигателями, или другими объектами в реальном времени. Другой широкой областью их применения в последнее время становится мобильная телефония, где ранее использовались два процессора — один обычный для управления функциями аппарата (дисплеем, клавиатурой), а другой для обработки голосовых сигналов (кодирование и т. д.).

Классификация ЦСП по назначению

В целом, по назначению ЦСП можно разделить на две группы:

- ЦСП общего назначения;

■ Проблемно-ориентированные ЦСП.

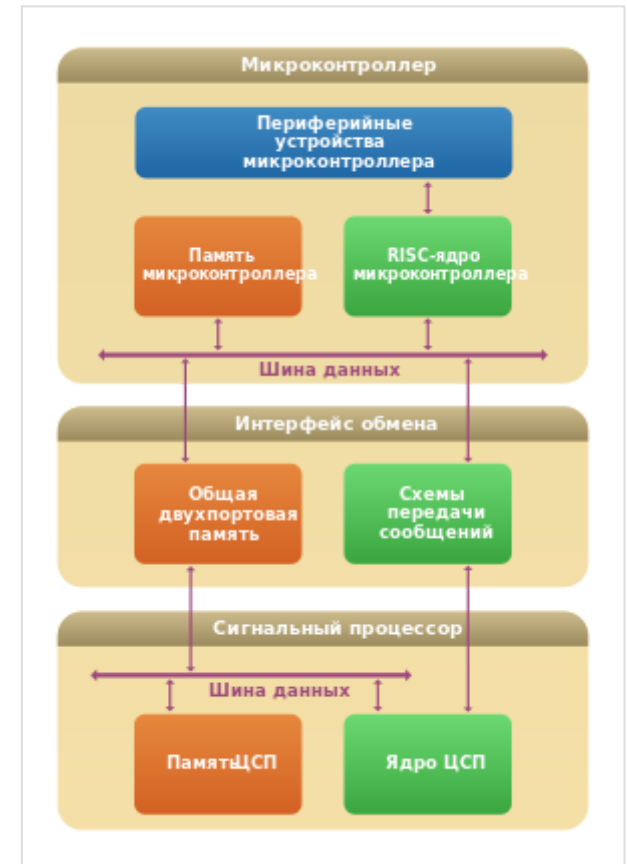
«Проблемная ориентация» обычно относится не к дополнительным командам, а к набору встроенных специализированных периферийных устройств. Например, ЦСП, предназначенные для управления электродвигателями, могут содержать на кристалле генераторы сигналов ШИМ, контроллеры локальной промышленной сети и т. д. Процессоры, используемые для обработки голосовых сигналов, часто содержат модули манипуляции разрядами (VMU) и сопроцессоры исправления ошибок. В цифровых фото- и видеокамерах применяются ЦСП с модулями кодирования/декодирования MPEG1, MPEG4, JPG, MP3, AAC и др.

Программирование ЦСП

Для программирования ЦСП обычно используют один из двух языков — ассемблер и C. Основные особенности ассемблеров ЦСП совпадают с аналогичными языками обычных микропроцессоров и, в целом, могут быть описаны как:

- Язык ассемблера является машинно-ориентированным, то есть каждое семейство процессоров имеет язык, отличающийся от языка других семейств;
- Одна команда на ассемблере обычно эквивалентна одной команде машинного языка;
- При программировании на ассемблере программисту доступны все ресурсы процессора и системы, что позволяет использовать их максимально эффективно;
- От программиста требуется хорошее знание архитектуры каждого конкретного процессора, с которым он работает, то есть требуемая квалификация персонала должна быть достаточно высокой;
- Создание и отладка программ на ассемблере — длительный трудоёмкий процесс, также требующий высокой квалификации.

С другой стороны, при использовании языков среднего и высокого уровня, в частности, C, можно заметно упростить и ускорить создание программ, но при этом ресурсы системы будут использоваться менее эффективно, по сравнению с программой, целиком написанной на ассемблере.



Блок-схема гибридного ЦСП

В реальности обычно используется подход, совмещающий достоинства как языков высокого уровня, так и эффективности программ на ассемблере. Выражается это в том, что стандартные библиотеки обычно создаются на ассемблере, как и критичные ко времени исполнения и объёму памяти части кода. В то же время вспомогательные модули могут создаваться на языке высокого уровня, ускоряя и упрощая разработку программной системы в целом.

Особенности ассемблеров ЦСП

К интересным особенностям ассемблеров ЦСП можно отнести следующее:

- Наличие двух форм записи многих команд — *мнемонической* и *алгебраической*. Мнемоническая форма аналогична записи команд для обычных микропроцессоров, например, `ADD dst, src`. Другая, алгебраическая, в ассемблерах стандартных микропроцессоров используется реже, в то время как на языке ЦСП упомянутая команда может быть записана в виде `dst = dst + src`. Обычно ассемблеры ЦСП понимают обе формы записи, но, например, ассемблеры *Analog Devices* и *Lucent Technologies* используют только алгебраическую запись.
- Средства организации стандартных структур, например, специальных аппаратных команд повторения одной команды или блока кода. При этом, в отличие от команд повторения обычных процессоров, ЦСП может пропускать цикл выборки кода повторяемой команды, что уменьшает время выполнения каждого повторения как минимум на 1 цикл шины, что при двухцикловой команде даёт двойной выигрыш по времени.

Совместимость внутри семейств ЦСП

Обычно ЦСП выпускаются семействами, и изделия внутри семейств имеют аналогичные языки ассемблера, или даже совместимы на уровне машинных кодов. Также внутри семейства обычно используются одинаковые наборы библиотек подпрограмм. Как и в обычных микропроцессорах, зачастую старшие модели ЦСП могут исполнять машинный код младших моделей, либо их ассемблер включает все команды младших моделей как подмножество собственного набора команд.

Отладка программ

Обычно отладка программ, написанных для ЦСП, производится с помощью специальных средств, включающих программные имитаторы и эмуляторы. В их состав также часто входят средства профилирования (измерения скорости выполнения блоков кода).

Примечания

1. Цифровые процессоры обработки сигналов: справочник. Под. ред. А. Г. Остапенко, М., Радио и связь, 1994.

2. *Солонина А. И., Улахович Д. А., Яковлев Л. А.* Алгоритмы и процессоры цифровой обработки сигналов. — СПб.: БХВ-Петербург, 2001. — 464 с. — ISBN 5-94157-065-1.

Литература

- *Солонина А. И., Улахович Д. А., Яковлев Л. А.* Алгоритмы и процессоры цифровой обработки сигналов. — СПб.: БХВ-Петербург, 2001. — 464 с. — ISBN 5-94157-065-1.
 - Применение цифровой обработки сигналов \ Под ред. Э. Опенгейма — МИР, 1980.
 - Цифровые сигнальные процессоры. Мир ПК, 5'93
 - Цифровые сигнальные процессоры фирмы Zilog и их применение. CHIPNEWS, № 2 (11) 1997 г.
 - Марков. С. Цифровые сигнальные процессоры. Книга 1. М.: Микроарт, 1996 г.
-

Источник — https://ru.wikipedia.org/w/index.php?title=Цифровой_сигнальный_процессор&oldid=126283007

Эта страница в последний раз была отредактирована 26 октября 2022 в 12:12.

Текст доступен по лицензии Creative Commons «С указанием авторства — С сохранением условий» (CC BY-SA); в отдельных случаях могут действовать дополнительные условия.

Wikipedia® — зарегистрированный товарный знак некоммерческой организации Фонд Викимедиа (Wikimedia Foundation, Inc.)