

Краткая история и основные термины

Современные центральные процессоры (ЦП), выполняемые в виде отдельных микросхем (чипов), реализующих все особенности, присущие данного рода устройствам, называют микропроцессорами. С середины 1980-х последние практически вытеснили прочие виды ЦП, вследствие чего термин стал всё чаще и чаще восприниматься как обыкновенный синоним слова «микропроцессор». (с 1980 года ЦП=микропроцессор)

Тем не менее, это не так: центральные процессорные устройства некоторых суперкомпьютеров даже сегодня представляют собой сложные комплексы, построенные на основе микросхем большой (БИС) и сверхбольшой (СБИС) степени интеграции.

Изначально термин *центральное процессорное устройство* описывал специализированный класс логических машин, предназначенных для выполнения сложных компьютерных программ. Вследствие довольно точного соответствия этого назначения функциям существовавших в то время компьютерных процессоров, он естественным образом был перенесён на сами компьютеры. Начало применения термина и его аббревиатуры по отношению к компьютерным системам было положено в 1960-е годы. Устройство, архитектура и реализация процессоров с тех пор неоднократно менялись, однако их основные исполняемые функции остались теми же, что и прежде.

Первые ЦП создавались в виде уникальных составных частей для уникальных, и даже единственных в своём роде, компьютерных систем. Позднее от дорогостоящего способа разработки процессоров, предназначенных для выполнения одной единственной или нескольких узкоспециализированных программ, производители компьютеров перешли к серийному изготовлению типовых классов многоцелевых процессорных устройств. Тенденция к стандартизации компьютерных комплектующих зародилась в эпоху бурного развития полупроводниковых элементов, мейнфреймов и миникомпьютеров, а с появлением интегральных схем она стала ещё более популярной.

Создание микросхем позволило ещё больше увеличить сложность ЦП с одновременным уменьшением их физических размеров. Стандартизация и миниатюризация процессоров привели к глубокому проникновению основанных на них цифровых устройств в повседневную жизнь человека. Современные процессоры можно найти не только в таких высокотехнологичных устройствах, как компьютеры, но и в автомобилях, калькуляторах, мобильных телефонах и даже в детских игрушках. Чаще всего они представлены микроконтроллерами, где помимо вычислительного устройства на кристалле расположены дополнительные компоненты (память программ и данных, интерфейсы, порты ввода/вывода, таймеры и др.). Современные вычислительные возможности микроконтроллера сравнимы с процессорами персональных ЭВМ десятилетней давности, а чаще даже значительно превосходят их показатели.

Архитектура фон Неймана

Большинство современных процессоров для персональных компьютеров в общем основаны на той или иной версии циклического процесса последовательной обработки данных, изобретённого Джоном фон Нейманом.

Дж. фон Нейман предложил схему построения компьютера в 1946 году.

Важнейшие этапы этого процесса приведены ниже. В различных архитектурах и для различных команд могут потребоваться дополнительные этапы. Например, для арифметических команд могут потребоваться дополнительные обращения к памяти, во время которых производится считывание операндов и запись результатов. Отличительной особенностью *архитектуры фон Неймана* является то, что инструкции и данные хранятся в одной и той же памяти. В отличие от *гарвардской архитектуры*, в которой программный код и данные хранятся в разной памяти. В *Гарвардской архитектуре*

невозможны многие методы программирования. Например, программа не может во время выполнения менять свой код; невозможно динамически перераспределять память между программным кодом и данными; зато гарвардская архитектура позволяет более эффективно выполнять работу в случае ограниченных ресурсов, поэтому она часто применяется во встраиваемых системах.

Этапы цикла выполнения:

1. Процессор выставляет число, хранящееся в [регистре счётчика команд](#), на [шину адреса](#) и отдаёт [памяти](#) команду чтения.
2. Выставленное число является для памяти [адресом](#); память, получив адрес и команду чтения, выставляет содержимое, хранящееся по этому адресу, на [шину данных](#) и сообщает о готовности.
3. Процессор получает число с шины данных, интерпретирует его как команду ([машинную инструкцию](#)) из своей [системы команд](#) и исполняет её.
4. Если последняя команда не является [командой перехода](#), процессор увеличивает на единицу (в предположении, что длина каждой команды равна единице) число, хранящееся в счётчике команд; в результате там образуется адрес следующей команды.

Данный цикл выполняется неизменно, и именно он называется *процессом* (откуда и произошло название устройства).

Во время процесса процессор считывает последовательность команд, содержащихся в памяти, и исполняет их. Такая последовательность команд называется программой и представляет алгоритм работы процессора. Очередность считывания команд изменяется в случае, если процессор считывает команду перехода, — тогда адрес следующей команды может оказаться другим. Другим примером изменения процесса может служить случай получения команды останова или переключение в режим обработки прерывания.

Команды центрального процессора являются самым нижним уровнем управления компьютером, поэтому выполнение каждой команды неизбежно и безусловно. Не производится никакой проверки на допустимость выполняемых действий, в частности, не проверяется возможная потеря ценных данных. Чтобы компьютер выполнял только допустимые действия, команды должны быть соответствующим образом организованы в виде необходимой программы.

Скорость перехода от одного этапа цикла к другому определяется [тактовым генератором](#). Тактовый генератор вырабатывает импульсы, служащие ритмом для центрального процессора. Частота тактовых импульсов называется тактовой частотой.

Конвейерная архитектура

Конвейерная архитектура (*pipelining*) была введена в центральный процессор с целью повышения быстродействия. Обычно для выполнения каждой команды требуется осуществить некоторое количество однотипных операций, например: выборка команды из [ОЗУ](#), дешифровка команды, адресация операнда в ОЗУ, выборка операнда из ОЗУ, выполнение команды, запись результата в ОЗУ. Каждую из этих операций сопоставляют одной ступени конвейера. Например, конвейер микропроцессора с архитектурой [MIPS-I](#) содержит четыре стадии:

- получение и декодирование инструкции,
- адресация и выборка операнда из ОЗУ,
- выполнение арифметических операций,
- сохранение результата операции.

После освобождения k -й ступени конвейера она сразу приступает к работе над следующей командой. Если предположить, что каждая ступень конвейера тратит единицу времени на свою работу, то выполнение команды на конвейере длиной в n ступеней займёт n единиц времени, однако в самом оптимистичном случае результат выполнения каждой следующей команды будет получаться через каждую единицу времени.

Действительно, при отсутствии конвейера выполнение команды займёт n единиц времени (так как для выполнения команды по-прежнему необходимо выполнять выборку, дешифровку и т. д.), и для исполнения m команд понадобится $n \cdot m$ единиц времени; при использовании конвейера (в самом оптимистичном случае) для выполнения m команд понадобится всего лишь $n + m$ единиц времени.

Факторы, снижающие эффективность конвейера:

1. Простой конвейера, когда некоторые ступени не используются (напр., адресация и выборка операнда из ОЗУ не нужны, если команда работает с регистрами).
2. Ожидание: если следующая команда использует результат предыдущей, то последняя не может начать выполняться до выполнения первой (это преодолевается при использовании внеочередного выполнения команд — out-of-order execution).
3. Очистка конвейера при попадании в него команды перехода (эту проблему удаётся сгладить, используя предсказание переходов).

Некоторые современные процессоры имеют более 30 ступеней в конвейере, что увеличивает производительность процессора, однако приводит к большому времени простоя (например, в случае ошибки в предсказании условного перехода). Не существует единого мнения по поводу оптимальной длины конвейера: различные программы могут иметь существенно различные требования.

Суперскалярная архитектура

Способность выполнения нескольких машинных инструкций за один такт процессора путем увеличения числа исполнительных устройств. Появление этой технологии привело к существенному увеличению производительности. В то же время существует определенный предел роста числа исполнительных устройств, при превышении которого производительность практически перестает расти, а исполнительные устройства простаивают. Частичным решением этой проблемы являются, например, технология Hyper Threading.

CISC-процессоры

Complex instruction set computer (CISC)— вычисления со сложным набором команд. Процессорная архитектура, основанная на усложнённом наборе команд. Типичными представителями CISC являются микропроцессоры семейства x86 (хотя уже много лет эти процессоры являются CISC только по внешней системе команд: в начале процесса исполнения сложные команды разбиваются на более простые микрооперации (МОП'ы), исполняемые RISC-ядром).

RISC-процессоры

Reduced instruction set computer (RISC)— вычисления с упрощённым набором команд (в литературе слово «reduced» нередко ошибочно переводят как «сокращённый»). Архитектура процессоров, построенная на основе упрощённого набора команд, характеризуется наличием команд фиксированной длины, большого количества регистров, операций типа регистр-регистр, а также отсутствием косвенной адресации. Концепция RISC разработана Джоном Коком (John Cocke) из IBM Research, название придумано Дэвидом Паттерсоном (David Patterson).

Упрощение набора команд призвано сократить конвейер, что позволяет избежать задержек на операциях условных и безусловных переходов. Однородный набор регистров упрощает работу компилятора при оптимизации исполняемого программного кода. Кроме того, RISC-процессоры отличаются меньшим энергопотреблением и тепловыделением.

Среди первых реализаций этой архитектуры были процессоры [MIPS](#), [PowerPC](#), [SPARC](#), [Alpha](#), [PA-RISC](#). В мобильных устройствах широко используются [ARM](#)-процессоры.

MISC-процессоры

Minimum instruction set computer (MISC)— вычисления с минимальным набором команд. Дальнейшее развитие идей команды Чака Мура, который полагает, что принцип простоты, изначальный для RISC-процессоров, слишком быстро отошёл на задний план. В пылу борьбы за максимальное быстродействие, RISC догнал и перегнал многие CISC процессоры по сложности. Архитектура MISC строится на **стековой вычислительной модели** с ограниченным числом команд (примерно 20-30 команд).

VLIW-процессоры

Very long instruction word (**VLIW**) — сверхдлинное командное слово. Архитектура процессоров с явно выраженным параллелизмом вычислений, заложенным в систему команд процессора. Являются основой для архитектуры [EPIC](#). Ключевым отличием от суперскалярных CISC-процессоров является то, что для них загрузкой исполнительных устройств занимается часть процессора (планировщик), на что отводится достаточно малое время, в то время как загрузкой вычислительных устройств для VLIW-процессора занимается компилятор, на что отводится существенно больше времени (качество загрузки и, соответственно, производительность теоретически должны быть выше). Примером VLIW-процессора является Intel [Itanium](#).

Многоядерные процессоры

Содержат несколько процессорных ядер в одном корпусе (на одном или нескольких кристаллах).

Процессоры, предназначенные для работы одной копии операционной системы на нескольких ядрах, представляют собой высокоинтегрированную реализацию мультипроцессорности.

Первым многоядерным микропроцессором стал [POWER4](#) от [IBM](#), появившийся в 2001 и имевший два ядра.

В октябре [2004 Sun Microsystems](#) выпустила двухъядерный процессор [UltraSPARC IV](#), который состоял из двух модифицированных ядер [UltraSPARC III](#). В начале 2005 был создан двухъядерный [UltraSPARC IV+](#).

[14 ноября 2005 года](#) Sun выпустила восьмиядерный [UltraSPARC T1](#), каждое ядро которого выполняло 4 [потока](#).

[5 января 2006 года](#) Intel представила первый двухъядерный процессор на одном кристалле Core Duo, для мобильной платформы

[10 сентября 2007 года](#) были выпущены в продажу нативные (в виде одного кристалла) четырёхъядерные процессоры для серверов [AMD Opteron](#), имевшие в процессе разработки кодовое название AMD Opteron Barcelona. [19 ноября](#) 2007 года вышел в продажу четырёхъядерный процессор для домашних компьютеров [AMD Phenom](#). Эти процессоры реализуют новую микроархитектуру K8L (K10).

В октябре [2007 года](#) в продаже появились восьмиядерные [UltraSPARC T2](#), каждое ядро выполняло 8 потоков.

В марте [2010 года](#) компания [AMD](#) выпустила первые в мире 12-ядерные серверные процессоры [Opteron 6100](#) архитектуры [x86](#).

На данный момент массово доступны двух-, четырёх- и шестиядерные процессоры, в частности [Intel Core 2 Duo](#) на 65-нм ядре Conroe (позднее на 45-нм ядре Wolfdale) и [Athlon 64 X2](#) на базе микроархитектуры K8. В ноябре 2006 года вышел первый четырёхъядерный процессор [Intel Core 2 Quad](#) на ядре Kentsfield, представляющий собой сборку из двух кристаллов Conroe в одном корпусе. Потомком этого процессора стал Intel Core 2 Quad на ядре Yorkfield (45 нм), архитектурно схожем с Kentsfield но имеющем больший объём кэша и рабочие частоты.

Компания AMD пошла по собственному пути, изготавливая четырёхъядерные процессоры единым кристаллом (в отличие от Intel, первые четырехъядерные процессоры которой представляют собой фактически склейку двух двухъядерных кристаллов). Несмотря на всю прогрессивность подобного подхода первый «четырёхъядерник» фирмы, получивший название AMD Phenom X4, получился не слишком удачным. Его отставание от современных ему процессоров конкурента составляло от 5 до 30 и более процентов в зависимости от модели и конкретных задач.

К 1-2 кварталу 2009 года обе компании обновили свои линейки четырёхъядерных процессоров. Intel представила семейство Core i7, состоящее из трёх моделей, работающих на разных частотах. Основными изюминками данного процессора является использование трёхканального контроллера памяти (типа DDR-3) и [технологии эмулирования](#) восьми ядер (полезно для некоторых специфических задач). Кроме того, благодаря общей оптимизации архитектуры удалось значительно повысить производительность процессора во многих типах задач. Слабой стороной платформы, использующей Core i7, является её чрезмерная стоимость, так как для установки данного процессора необходима дорогая материнская плата на чипсете [Intel X58](#) и трёхканальный набор памяти типа [DDR3](#), также имеющий на данный момент высокую стоимость.

Компания AMD в свою очередь представила линейку процессоров Phenom II X4. При её разработке компания учла свои ошибки: был увеличен объём кэша (явно недостаточный у первого «Фенома»), а производство процессора было переведено на 45-нм техпроцесс, позволивший снизить тепловыделение и значительно повысить рабочие частоты. В целом, AMD Phenom II X4 по производительности стоит вровень с процессорами Intel предыдущего поколения (ядро Yorkfield) и весьма значительно отстаёт от Intel Core i7. Однако, принимая во внимание умеренную стоимость платформы на базе этого процессора, его рыночные перспективы выглядят куда более радужно, чем у предшественника.