

1. Введение

1.1. Введение в сетевые академии Cisco

В области информационных технологий (IT - Information Technology) можно привести некоторые признаки начинающегося кризиса в области трудовых компетентных ресурсов: в настоящее время во многих средних и крупных компаниях существуют незаполненные вакансии в сфере современных технологий. Большинство технологических компаний собираются в ближайшие несколько лет расширить свой штат IT персонала. Компании называют нехватку квалифицированных работников IT одним из барьеров, препятствующих росту бизнеса.

Следовательно, перед учебными заведениями и в частности Компьютерной Академией ШАГ ставится задача дать студентам тот уровень компетентности, который позволит им достичь профессионального успеха в XXI столетии.

Учебные программы Сетевой Академии Cisco, предлагаемые нами в процессе обучения, обеспечивает получение востребованных в сфере интернет-технологий навыков проектирования, строительства и обслуживания сетей, а также включает основы различных операционных систем, информационных технологий, прокладки сетей передачи голоса и данных, основы безопасности сети и основы беспроводных технологий. Материалы программы регулярно обновляются, что позволяет слушателям получать актуальные знания, отвечающие требованиям сегодняшнего дня.

Назначение программы предлагаемой КА“ШАГ” для подготовки специалистов - фундаментальная и компетентная подготовка специалистов в области IT.

Программа Сетевой Академии обеспечивает доступ к информационным Интернет-ресурсам и включает в себя средства онлайн-тестирования и проверки успеваемости, практические лабораторные занятия, консультации и поддержку преподавателей, а также подготавливает слушателей к получению сертификатов международного образца.

Мы с вами начинаем обучение по курсу *ITEssentialsI - PC Hardware and Software, Apparantное и программное обеспечение ПК*, который финансируется компанией Hewlett-Packard, и является введением в информационные технологии и технологии передачи данных.

В течение курса вы приобретете навыки и знания в: сборка компьютера: установка различных составляющих, установка и конфигурирование семейства ОС Windows, установка периферийных и мультимедийных устройств, архитектура локальных сетей, сетевые протоколы, модель OSI и служебные программы TCP/IP, подключение компьютера к локальной сети и Интернету.

Но хотелось бы особенно отметить, что программа *ITEssentialsI* выходит за рамки традиционных компьютерных учебных программ, она построена таким образом, что бы помочь обучающимся получить исчерпывающий набор знаний и навыки в области устройства персонального компьютера, близко соответствующей реальной обстановке, в которой приходится работать при обслуживании информационных систем.

После окончания курса вы уже сможете продемонстрировать свою компетентность, работая системными администраторами в небольших компаниях, консультантами по использованию программного и аппаратного оборудования и в других специальностях.

Следует также отметить, что по всем пройденным курсам, слушателям выдается именной сертификат Cisco об окончании курса.

1.2 Процедура регистрации на сайте Cisco Network Academy.

А теперь давайте попробуем пройти регистрацию на WEB ресурсе сетевой академии Cisco. На первом занятии, вам необходимо было сообщить вашему преподавателю свой почтовый адрес. На который вы получите письмо, темой данного письма будет запись, вида:

Дата: 09 Мая 2008 09:46:22
Тема: Academy Connection Account Created

В самом письме нас интересует текст

* To begin, point your browser to <http://cisco.netacad.net> (remember to bookmark this page) and login using:
Initial ID: 5675312
Password: 301598

В котором указано, что для дальнейшей регистрации, вам необходимо посетить сайт <http://cisco.netacad.net>



Где нас с вами в первую очередь будут интересовать поля **Username** и **Password** в которых вам и необходимо будет указать, как пример **Username – 5675312 (Initial ID)** в **password – 301598**, то есть получим приблизительно следующее

Username:

Password:
 GO

- New User? [Learn more.](#)
- Forgot your [Password?](#)

***Замечание:** Данные логин и пароль являются одноразовыми, и используются только для того что бы вы могли зайти на сайт, и получить доступ к процедуре регистрации.*

Если вы все правильно сделали, тогда вам откроется страничка, на которой вы увидите



Это ничто иное как лицензионное соглашение, внимательно с ним ознакомившись, и если у вас нет возражений, тогда в самом его конце нажимаем кнопку **I Accept**.

Следующим нашим шагом будет заполнение регистрационной формы, пожалуйста уделите этому особое внимание, что бы в дальнейшем не было проблем с вашей сертификацией в будущем. И хотелось бы заметить, что если у вас есть загран. паспорт, укажите фамилию на английском, как в паспорте.

Замечание: Обратите внимание, на критерии безопасности для пароля, а в частности, пароль должен соответствовать трем условиям из перечисленных, а это:

- Содержать большие и маленькие буквы
- Цифры
- Специальные символы `!$%^&*()_+|~-=\`{}[]:;';<>?.,/`
- Не должен содержать ваши Имя и Фамилию

А также, хотелось бы обратить ваше внимание, что необходимо обязательно заполнить все регистрационные элементы помеченные * - звездочкой.

Если регистрация прошла успешно, вы увидите следующую страничку.

Где нас с вами в первую очередь будут интересовать ваш класс в который вы зачислены (отмечено цифрой 1) и материалы курса (отмечено цифрой 2). Если по какой либо причине, у вас появилось данное окно, однако в нем не отображается ваш класс, в таком случае выйдите с сайта, и попробуйте войти повторно.

Теперь поподробнее о материалах курса, и а том как получить к ним доступ. Для этого выберем справа в **Resources – Course Materials**, после чего нам откроется следующее окно

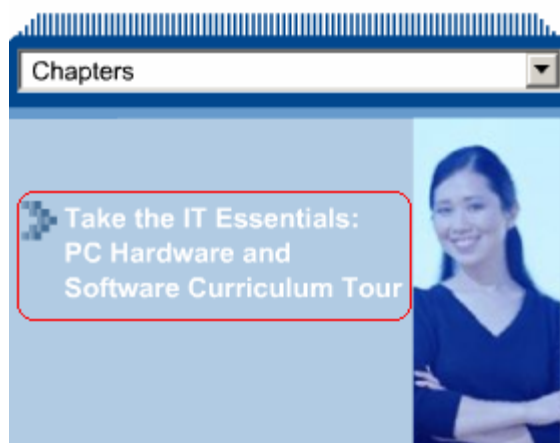
Выбрав язык, название курса и версию материала (а мы будем пользоваться последней версией документации 4.0), смело нажимаем **Submit**.

Далее нам предлагается выбрать на каком языке мы все же хотим прочитать материалы, конечно же в этом случае мы выбираете, тот язык на котором материал вам будет понятнее, однако мы бы вам рекомендовали читать английскую версию материалов курса, так как в данном случае вы приобретете хороший словарный запас технического английского языка, обращаясь к русской версии только при трудностях перевода.

Course Materials

Course - Version	View [Last Updated]
Russian	
IT Essentials: PC Hardware and Software - 4.0	Launch [04/17/2008]
English	
IT Essentials: PC Hardware and Software - 4.0	Launch [06/02/2007]

После чего нам открывается материал, с возможностью выбора главы с которой мы бы хотели ознакомиться, если же у вас возникают проблемы с пониманием интерфейса курса, в таком случае просмотрите, дополнительный материал, о том как правильно пользоваться интерфейсом,



Остается только пожелать вам удачи, при освоении материалов курса

1.3 Рекомендации по материалам Cisco

Цель нашего сегодняшнего с вами занятия познакомиться с основными принципами и технологиями при рассмотрении функционирования Оперативной памяти и Центрального процессора нашего компьютера. Но для этого очень бы хотелось, для большего понимания принципов работы этих устройств что бы вы рассмотрели материал предлагаемый компанией Cisco по программе *ITEssentials - PC Hardware and Software*, а в частности.

Глава 1. Основы работы с персональным компьютером.

Уделив особое внимание разделу

1.2 Описание компьютерной системы – уделив особое внимание процессу выбора корпуса, исходя из физическими размерами блока питания и форм фактора материнской платы.

1.3 Знакомство с названиями, предназначением и характеристиками корпусов и блоков питания – рассмотрев цветовую маркировку проводов электропитания, используемым разъемам и процессу расчета требуемого блока питания мощности которого будет достаточно для электропитания всех компонентов.

1.4.1 Знакомство с названиями, предназначением и характеристиками материнских плат – рассмотрев существующие Форм-факторы материнских плат.

1.4.2 Объяснение названий, предназначения и характеристик ЦП – уделив внимание типам Центрального процессора а также спецификации сокетов

1.4.4 Знакомство с названиями, предназначением и характеристиками ПЗУ и ОЗУ – ознакомившись с типами памяти их назначениями.

А также

Глава 11. Углубленное изучение персональных компьютеров

11.3 Описание ситуаций, требующих замены компонентов компьютера

11.4 Обновление и настройка компонентов персонального компьютера и периферийных устройств

Замечание: Хотелось бы отметить, что содержимое глав I и II необходимо прочитать полностью, и усвоить весь изложенный в них материал, но главам отмеченным выше необходимо уделить особое внимание, что бы не возникало затруднений с освоением дальнейших материалов курса.

2. Оперативная память

2.1. Общие принципы обработки данных.

При ознакомлении с материалами Cisco Network Academy по курсу *ITEssentials* вы обзорно уже ознакомились с основными компонентами, из которых состоит современный компьютер. Давайте теперь поговорим более детально о том, каким образом организована передача данных между ними. На рисунке 1.1 представлена типовая блок-схема современного компьютера. Давайте разберем ее подробнее.

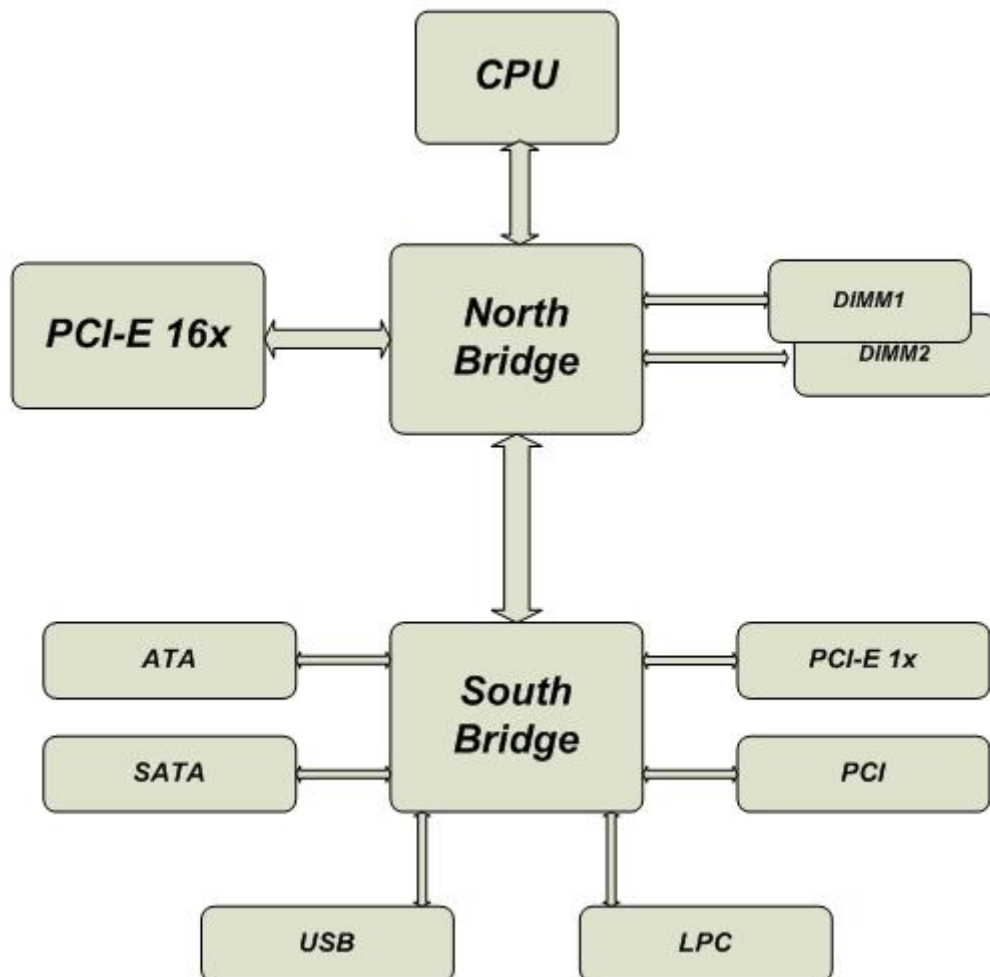


Рисунок 1.1 – Блок схема подключения узлов к набору системной логики.

Северный мост является основным компонентом в наборе системной логики, соединяющим наиболее быстрые узлы компьютера - процессор, память и специализированную шину PCI-Express между собой.

Как вы может уже знаете, процессор (за редкими исключениями) исполняет не программы, написанные на каком-нибудь языке программирования (один из которых, вы, возможно, даже знаете), а некий «машинный код». То есть командами для него являются последовательности байтов, находящихся в памяти компьютера. Иногда команда может быть равна одному байту, иногда она занимает несколько байт. Там же, в основной памяти (ОЗУ, RAM) находятся и данные. Они могут находиться в отдельной области, а могут и быть «перемешаны» с кодом. Различие между кодом и данными состоит в том, что данные — это то, над чем процессор производит какие-то операции. А код — это команды, которые ему сообщают, какую именно операцию он должен произвести. Для упрощения, мы можем представить себе программу и ее данные в виде последовательности байтов некой конечной длины, располагающуюся непрерывно (не будем усложнять) в общем массиве памяти. Например, у нас есть массив памяти длиной в 1'000'000

байт, а наша программа (вместе с данными) — это байты с номерами от 1000 до 20'000. Прочие байты — это другие программы или их данные, или просто свободная память, не занятая ничем полезным.

Таким образом, «машинный код» — это команды процессора, располагающиеся в памяти. Там же располагаются и данные. Для того чтобы исполнить команду, процессор должен прочитать ее из памяти. Для того чтобы произвести операцию над данными, процессор должен прочитать их из памяти, и, возможно, после произведения над ними определенного действия, записать их обратно в память в обновленном (измененном) виде

То есть получается, что как команды, так и данные, попадают в процессор из оперативной памяти. На самом деле всё немного сложнее. В большинстве современных систем (то есть компьютеров), процессор как устройство к памяти обращаться вообще не может, так как не имеет в своем составе соответствующих узлов. Поэтому он обращается к «промежуточному» специализированному устройству, называемому контроллером памяти, а уже тот, в свою очередь — к микросхемам ОЗУ, размещенным на модулях памяти. Роль контроллера ОЗУ, таким образом, проста: он служит своего рода «мостом» между памятью и использующими ее устройствами (кстати, к ним относится не только процессор, но об этом — чуть позже). Как правило, контроллер памяти входит в состав чипсета (северного моста из набора системной логики) — набора микросхем, являющегося основой системной платы. От быстродействия контроллера во многом зависит скорость обмена данными между процессором и памятью, это один из важнейших компонентов, влияющих на общую производительность компьютера.

Любой процессор, в свою очередь, обязательно оснащён процессорной шиной, которую принято называть FSB (Front Side Bus). Эта шина служит каналом связи между процессором и всеми остальными устройствами в компьютере: памятью, видеокартой, жёстким диском, и так далее. Впрочем, об этом мы уже с вами знаем, между собственно памятью и процессором находится контроллер памяти. Соответственно: процессор посредством FSB связывается с контроллером памяти, а уже тот, в свою очередь, по специальной шине (назовём её, не мудрствуя лукаво, «шиной памяти») — с модулями ОЗУ на плате. Однако, повторимся: поскольку «внешняя» шина у классического процессора всего одна, она используется не только для работы с памятью, но и для общения процессора со всеми остальными устройствами.

Но стоит заметить, что компания AMD о новом подходе к данонной схеме передачи данных из оперативной памяти в процессор, и состоит он в том, что её процессоры с архитектурой AMD64 (и микроархитектурой, которую условно принято называть «K8») оснащены множеством «внешних» шин. При этом одна или несколько шин HyperTransport служат для связи со всеми устройствами кроме памяти, а отдельная группа из одной или двух (в случае двухканального контроллера) шин — исключительно для работы процессора с памятью. То есть проще говоря AMD интегрировала контроллер оперативной памяти внутрь процессора. Преимущество интеграции контроллера памяти, очевидно: «путь от ядра до памяти» становится заметно «короче», что позволяет работать с ОЗУ быстрее. Правда, имеются у данного подхода и недостатки. Так, например, если ранее устройства типа жёсткого диска или видеокарты могли работать с памятью через выделенный, независимый контроллер — то в случае с архитектурой AMD64 они вынуждены работать с ОЗУ через контроллер, размещённый на процессоре. Так как CPU в данной архитектуре является единственным устройством, имеющим прямой доступ к памяти.

Де-факто, в противостоянии «внешний контроллер vs. интегрированный», сложился паритет: с одной стороны, на данный момент AMD является единственным производителем десктопных процессоров с интегрированным контроллером памяти, с другой — компания вроде бы вполне довольна этим решением, и не собирается от него отказываться. С третьей — Intel тоже не собирается отказываться от внешнего контроллера, и вполне довольна «классической схемой», проверенной годами.

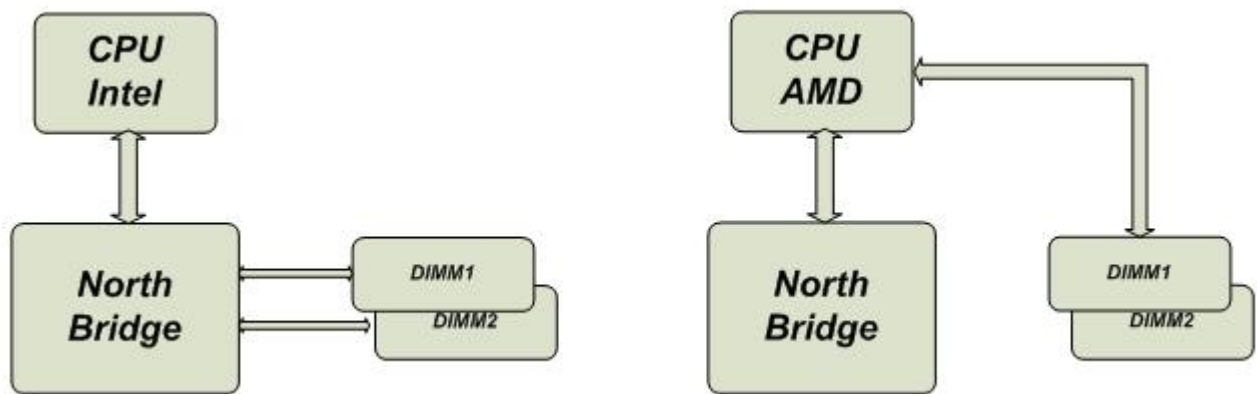


Рисунок 1.2 – Схема взаимодействия, Процессор – Оперативная память.

Отсюда можно сделать очевидный вывод, начинать анализ эффективности организации системы стоит с "ядра" – связки **процессор – память - чипсет**. В принципе можно начать изучение с любого из этих компонентов, однако сегодня, пожалуй, наиболее удобно было бы начать с памяти. Это и будет нашей сегодняшней темой!

2.2. Назначение Оперативной памяти.

Из материалов Cisco вы узнали что такое память, зачем она нужна, и как зависит производительность компьютера от её объёма. Разберем общие сведения об оперативной памяти.

Оперативная память - это рабочая область для процессора компьютера, в которой во время работы компьютера хранятся программы и данные. Оперативная память часто рассматривается как временное хранилище, потому что данные и программы в ней сохраняются только при включенном компьютере или до нажатия кнопки сброса (reset).

Перед выключением или нажатием кнопки сброса все данные, подвергнутые изменениям во время работы, необходимо сохранить на запоминающем устройстве, которое может хранить информацию постоянно (обычно это жесткий диск). При новом включении питания сохраненная информация опять может быть загружена в память.

Иногда люди путают оперативную память с памятью на диске, поскольку емкость устройств памяти обоих типов выражается в одинаковых единицах - мега- или гигабайтах. Попробуем объяснить связь между оперативной памятью и памятью на диске с помощью следующей простой аналогии.

Представьте себе небольшой офис, в котором некий сотрудник обрабатывает информацию, хранящуюся в картотеке. В нашем примере шкаф с картотекой будет выполнять роль жесткого диска системы, где длительное время хранятся программы и данные. Рабочий стол будет представлять оперативную память системы, которую в текущий момент обрабатывает сотрудник, - его действия подобны работе процессора. Он имеет прямой доступ к любым документам, находящимся на столе. Однако, прежде чем конкретный документ окажется на столе, его необходимо отыскать в шкафу. Чем больше в офисе шкафов, тем больше документов можно в них хранить, а если рабочий стол достаточно велик, можно одновременно работать с несколькими документами.

Добавление к системе жесткого диска подобно установке еще одного шкафа для хранения документов в офисе - компьютер может постоянно хранить большее количество информации. Увеличение объема оперативной памяти в системе подобно установке большего рабочего стола - компьютер может работать с большим количеством программ и данных одновременно.

Впрочем, есть одно различие между хранением документов в офисе и файлов в компьютере:

когда файл загружен в оперативную память, его копия все еще хранится на жестком диске. Обратите внимание: поскольку невозможно постоянно хранить файлы в оперативной памяти, все измененные после загрузки в память файлы должны быть вновь сохранены на жестком диске

перед выключением компьютера. Если измененный файл не будет сохранен, то первоначальная копия файла на жестком диске останется неизменной.

А теперь о материальном. В современный компьютер оперативная память устанавливается с помощью специальных модулей в соответствующие разъемы на материнской плате. И, естественно, мы с Вами изучим, как эти модули называются, чем отличаются. Но модуль памяти - это всего лишь договоренность о форме и размерах маленькой платки с припаянными чипами и о разъеме, куда она устанавливается. Ведь в первую очередь модуль состоит из микросхем памяти, и именно параметры микросхем (их архитектура и быстродействие) определяют эффективность того или иного модуля. Поэтому давайте сразу договоримся, что мы, с одной стороны, будем изучать типы памяти, т.е. логику организации тех или иных типов чипов памяти, их быстродействие и эффективность. А с другой стороны - в компьютер устанавливаются именно модули, и мы изучим внешний вид и параметры модулей, состоящих из чипов (микросхем) памяти конкретного типа (архитектуры).

2.3. Классификация памяти.

Итак, начнем с изучения типов оперативной памяти. Однако давайте напомним сами себе, какие же типы памяти вообще существуют, основные сведения о которых вы смогли почерпнуть из главы 1.4.4 курса по программе *ITEssentials*.

В современных компьютерах используются запоминающие устройства двух основных типов, которые в свою очередь делятся на подвиды:

ROM (Read Only Memory). Постоянное запоминающее устройство — ПЗУ. Именно поэтому такая память используется только для чтения данных. ROM также часто называется *энергонезависимой памятью*, потому что любые данные, записанные в нее, сохраняются при выключении питания. Поэтому в ROM помещаются команды запуска ПК, т.е. программное обеспечение, которое загружают систему.

RAM (Random Access Memory). Память с произвольным доступом. Особенность данного вида памяти заключается в том что при необходимости всегда можно получить доступ к данным хранящейся в любой ячейки памяти.

DRAM (Dynamic Random Access Memory). Динамическое запоминающее устройство с произвольным порядком выборки. Ячейки памяти в микросхеме DRAM — это крошечные конденсаторы, которые удерживают заряды. Именно так (наличием или отсутствием зарядов) и кодируются биты.

SRAM (Static RAM). Статическая оперативная память. Она названа так потому, что, в отличие от динамической оперативной памяти (DRAM), для сохранения ее содержимого не требуется периодической регенерации. Но это не единственное ее преимущество. SRAM имеет более высокое быстродействие, чем DRAM, и может работать на той же частоте, что и современные процессоры.

А теперь давайте попробуем классифицировать виды электронной памяти. По устройству и принципам работы, можно выделить следующие классификации:

- динамическая или статическая.
- асинхронная или синхронная.
- энергозависимая или энергонезависимая.

Начнем с первой классификации:

Динамическая память, или DRAM, Dynamic Random Access Memory.

Информация хранится в ячейке, состоящей из конденсатора, доступ к которому управляется транзистором. Наличие или отсутствие заряда этого конденсатора, собственно и

представляет хранимую информацию (0 или 1). Именно использование конденсатора делает этот вид памяти относительно дешевым и простым в изготовлении, но в тоже время приводит к тому, что увеличивается время доступа – во первых конденсатор не может мгновенно зарядиться или разрядиться, по этой же причине динамической памяти не доступны высокие частоты работы.

Во вторых при чтении конденсатор разряжается, и к тому же конденсатор не может долго хранить заряд, т.е. он постепенно разряжается. Поэтому ячейки динамической памяти необходимо постоянно обновлять (регенерировать), на что также тратится драгоценное время ожидания процессора.

Статическая память, или SRAM, Static Random Access Memory.

В отличии от динамической памяти, ячейка, хранящая информацию, представляет из себя электронный переключатель – триггер, который хранит своё значение, пока есть питание. Такой элемент позволяет быстро может изменять своё значение, что позволяет использовать его на очень высоких частотах, и не требует регенерации, что даёт очень качественные показатели времени доступа. К сожалению, такой вид памяти сложен в изготовлении, и очень дорого стоит. Какой вывод можно сделать по этим видам памяти?

DRAM – дешево, но медленно, SRAM – дорого, но быстро. Поэтому у DRAM и SRAM разные сферы применения, DRAM применяется в качестве оперативной памяти, а SRAM – в качестве так называемой кэш-памяти, промежуточной быстродействующей памяти между процессором и оперативной памятью. DRAM, как оперативную память мы рассмотрим сегодня, а о роли SRAM поговорим чуть позже когда будем обсуждать процессоры.

Вторая классификация:

Асинхронная память.

Такая память выдаёт данные с меньшей частотой, чем частота шины, на которой она работает. Типичный пример такой памяти – классическая DRAM, где на ожидание каждого бита информации уходило около 5 тактов работы шины. Как вы понимаете, такая память имеет большую латентность, но в то время когда она была популярна, процессоры с которыми она применялась тоже не отличались быстродействием. Позже появились модифицированные виды асинхронной DRAM позволяющие получить меньшее значение латентности, но это уже история развития памяти, и о ней вы сможете ознакомиться чуть позже.

Синхронная память.

Этот вид памяти обменивается данными с контроллером на той же частоте, на которой работает шина памяти. Типичный пример – **SDRAM, Synchronous DRAM**. Время доступа к такой памяти определяется частотой, на которой она работает. Эффективность SDRAM намного выше, чем у ее предшественников. Во-первых, дело в том, что схема чтения у SDRAM намного эффективнее, чем у устаревших видов памяти. Что позволяет получить более высокую скорость работы связи

Процессор – Оперативная память.

И наконец, третья классификация:

Энергозависимая.

Такой вид памяти для хранения информации требует постоянное электрическое питание, т.е. она работает только в то время, когда включено питание компьютера. Все рассмотренные выше виды памяти являются именно энергозависимыми.

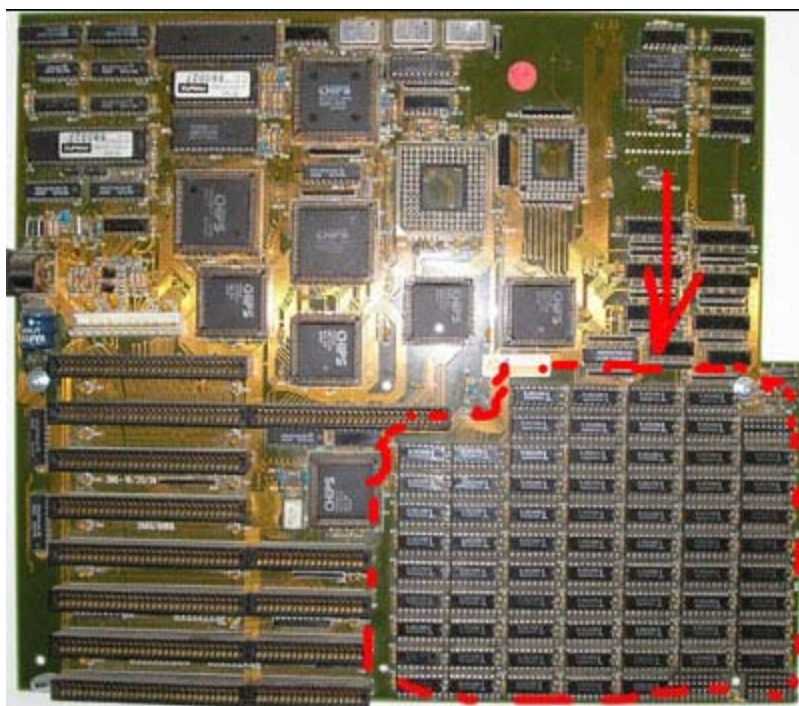
Энергонезависимая.

В отличии от предыдущего вида памяти, энергонезависимая память может хранить информацию даже при отсутствии электрического питания. Существует несколько видов энергонезависимой памяти. Классический представитель - **ROM, Read Only Memory**. Память только для чтения, информация в такие чипы записывалась при их изготовлении. Когда-то она использовалась для хранения BIOS, но из-за невозможности перезаписи, была заменена программируемыми видами постоянной памяти. Последний из таких видов – **EEPROM**, электрически стираемая программируемая постоянная память, которую мы знаем как **Flash**-память. Flash память может хранить информацию без обновления около 10 лет, и в настоящее время используется не только как чип для хранения

BIOS материнской платы, но и в качестве переносного накопителя информации, например USB Flash drive. Также разрабатываются новые виды памяти, которые смогли бы выступить в роли оперативной, и в тоже время были энергонезависимыми. Например в конце 90-х годов прошлого столетия, была разработана память **FeRAM** – ферроэлектрическая память, по своим скоростным характеристикам близкая к оперативной, но пока что всё равно медленней, и в тоже время может хранить информацию без дополнительного питания несколько лет. На этом рассмотрение видов памяти можно остановить, тем более что вы уже об этом слышали просматривая раздел **1.4.4** по курсу Cisco ITEssentials, и кроме того примите во внимание – прогресс не стоит на месте, и каждый день появляется что-нибудь новое, чего в урок мы поместить не можем, так что следите за новостями мира информационных технологий, и вы будете всегда в курсе событий

2.4. Виды модулей памяти.

2.4.1. Устаревшие модули памяти (не обязательно)



Настала пора рассмотреть, каким же образом оперативная память устанавливается в компьютеры. В первых PC память устанавливалась чипами непосредственно на материнскую плату, до определённого момента такой способ удовлетворял и пользователей, и производителей материнских плат, однако позднее возникла необходимость в создании более гибкого способа установки памяти на материнские платы. Т.е. надо было предоставить сборщикам компьютеров, или самим пользователям возможность выбора объёма памяти, тем более учитывая, что цены на оперативную память в то время (1 Мбайт около 40 долларов США). Представьте себе материнскую плату

(см. рисунок), на которой 2 Мбайта памяти устанавливается 72(!) микросхемами (на рисунке выделено), и в описании платы написано, что память можно расширить до 16-ти Мбайт, заменив установленные чипы на более ёмкие. Сам процесс такой замены для многих может показаться кошмаром. И тогда производители компьютерного оборудования решили использовать для установки памяти небольшие платы с припаянными чипами, которые устанавливались в специальные разъёмы в материнской плате. Эти платы и называются модулями.

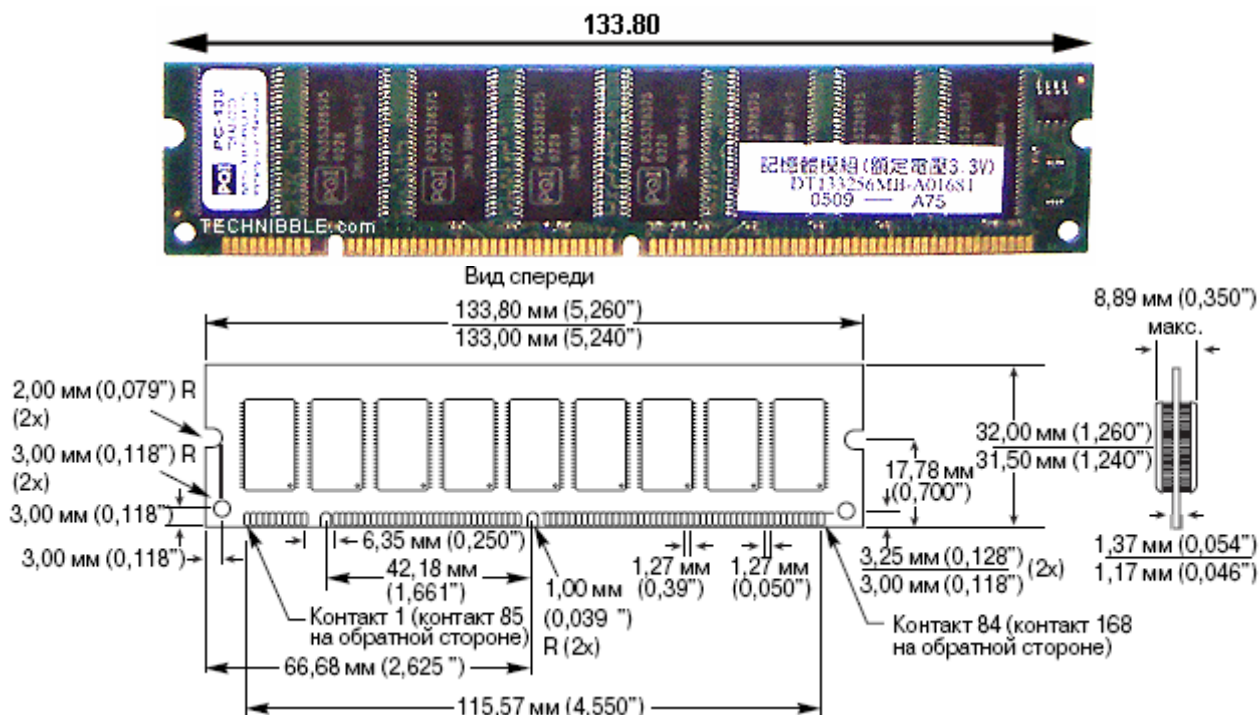
Итак, давайте рассмотрим типы модулей, которые применялись когда-либо, и применяются сейчас в персональных компьютерах:

SIMM. Single Inline Memory Module, модуль памяти с односторонним расположением контактов. Хотя контакты на таких модулях были с обеих сторон, эти контакты были замкнуты между сторонами, и поэтому модуль назывался односторонним. Выпускались модули SIMM в двух модификациях – SIMM 30 pin и SIMM 72 pin.



SIMM 72 pin – в основном применялись с FPM и EDO DRAM. Разрядность они имели уже большую – 32 бита. Такие модули применялись в 486-ых и Pentium-системах. Стало быть, в 486 системах, у которых ширина шины обмена процессор - память составляла 32 бита, банком памяти являлся единственный модуль памяти SIMM 72 pin, а в Pentium-системах, у которых разрядность шины памяти 64 бита - 2 модуля SIMM 72 pin.

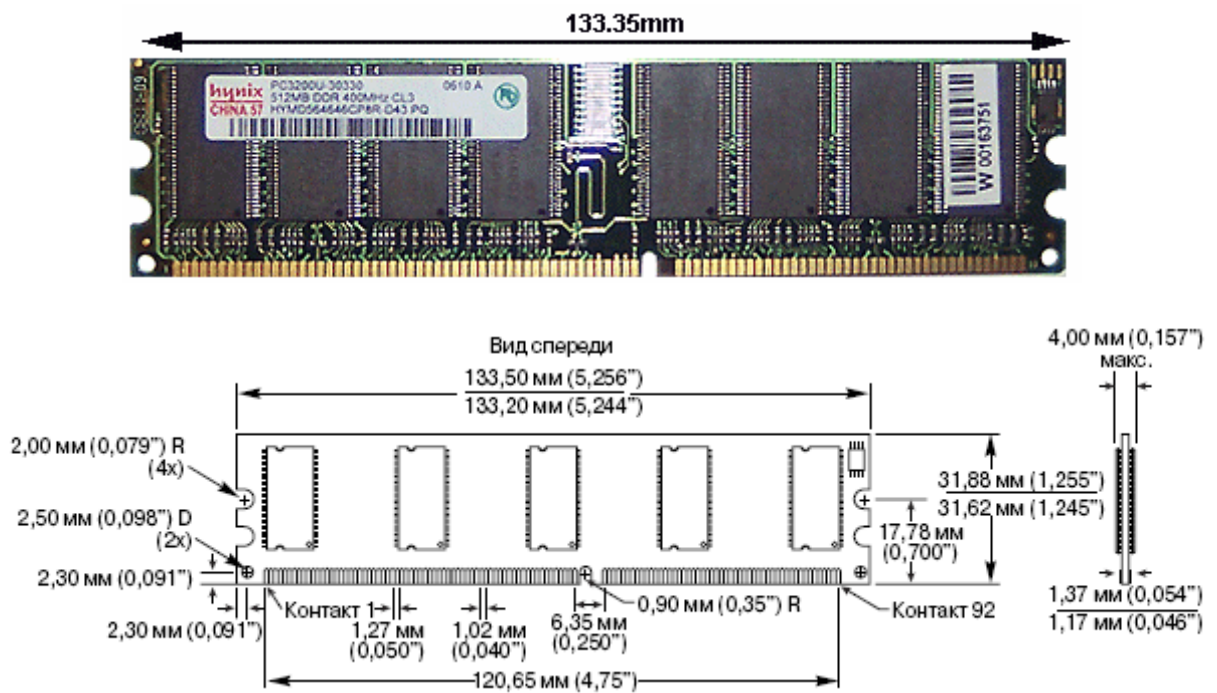
DIMM. Double Inline Memory Module, модуль памяти с двухсторонним расположением контактов. В этих модулях контакты разных сторон не замкнуты между собой. Разрядность таких модулей – 64 бита, и использование единственного модуля DIMM позволяет закрыть шину памяти - процессор для процессора Pentium, как, впрочем, и для любого современного процессора. Существует несколько разновидностей таких модулей:



DIMM 168 pin. Применялись с памятью SDRAM, поэтому такие модули называли просто DIMM SDRAM. Имели 2 ключа (прорези на плате между контактами. см. фото), предназначенные для защиты от неправильной установки модуля, и несущие некоторую информационную нагрузку (например, напряжение питания модуля). Появились они в то время, когда популярным процессором был Pentium, и применялись позднее с Pentium II, Pentium III и их аналогами. В настоящее время модули DIMM SDRAM можно найти только на компьютерных "барахолках". Вместе с этими модулями появилось понятие "маркировки" модуля памяти. Что это такое? Действительно, ведь Вы устанавливаете в компьютер не чипы, а модули, и умение читать маркировку чипа совсем не необходимо для сборки (или подбора комплектующих). Как же маркируются модули памяти DIMM SDRAM? А очень просто. Маркируются они следующим образом: PCxxx, где xxx - частота, на которой сертифицирован работать модуль (возможно что чипы, модуль составляющие, могут работать и на более высоких частотах). Соответственно, существует всего 3 спецификации DIMM SDRAM:

- PC66** - DIMM SDRAM, предназначенный для работы на частоте, не превышающей 66 МГц;
- PC100** - DIMM SDRAM, предназначенный для работы на частоте, не превышающей 100 МГц;
- PC133** - DIMM SDRAM, предназначенный для работы на частоте, не превышающей 133 МГц.

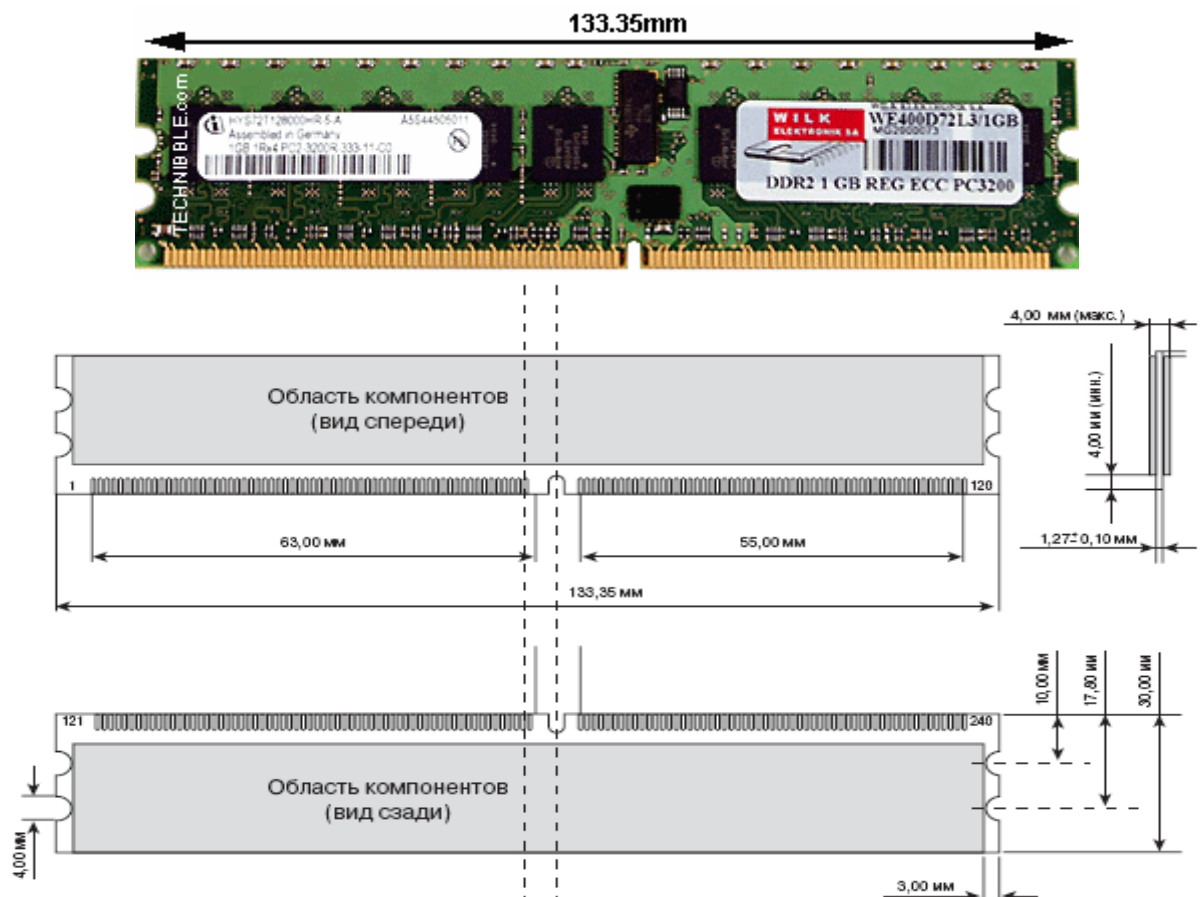
Естественно, модули, предназначенные для более высоких частот, могут без проблем применяться и на низших частотах.



DIMM 184 pin. Применяются с памятью DDR SDRAM, соответственно тоже назывались DIMM DDR SDRAM. В отличие от DIMM 168 pin, такие модули имеют всего 1 ключ, защищающий их от неправильной установки (см. рисунок). Эти модули памяти в настоящее время являются наиболее популярными сейчас, т.к. применяются в системах с процессорами Pentium 4 и различными модификациями Athlon.

Маркируются модули DIMM DDR SDRAM максимальной пропускной способностью, которую могут обеспечить. К расчету пропускной способности этих модулей добавляется ещё и коэффициент – 2, так как данные передаются в режиме DDR – 2 бита за 1 такт.

2.4.2. Современные модули памяти



DIMM 244 pin. Применяются с памятью DDRII SDRAM, тоже имеют 1 ключ, и внешне очень похожи на модули DIMM 184 pin, однако благодаря иному расположению ключа перепутать эти модули не получится. В настоящее время модули DDRII постепенно насыщают рынок памяти, при этом, необходимость применения таких модулей определяется выбором процессора и материнской платы (чипсета).

Основной причиной создания таких модулей послужила необходимость в увеличении

пропускной способности оперативной памяти, при этом используя более низкие частоты работы шины памяти, ведь чем больше частота работы, тем больше энергии уходит на поддержание работы такой памяти. Маркируются модули DDRII SDRAM также пропускной способностью, но при этом в их маркировке принято использовать обозначение PC2-XXXX, что бы избежать путаницы при маркировке модулей, чья пропускная способность равна пропускной способности более старых видов памяти.

Rambus

Кроме рассмотренных видов модулей, в конце 90-х, а конкретно в 1999 году фирма Intel попытались ввести на рынок памяти принципиально новую, более производительную память чем существовавшие в то время (PC100 SDRAM). Речь идет о памяти **Rambus, DR DRAM (Direct Rambus DRAM) или просто RDRAM.** До применения в PC, эта память применялась в игровых приставках и высокопроизводительных графических станциях SGI. Intel решили, что эта память должна стать основным видом памяти и для персональных компьютеров, и выпустили её на рынок с процессорами Pentium III. В чём причина такого решения?

Естественно, что на пути разработки высокоскоростных интерфейсов есть два решения: увеличивать частоту работы шины и увеличивать ширину шины. Разумеется, если можно было бы увеличивать эти два параметра так, как хочется производителю, то мы сегодня имели бы память, работающую на огромных частотах и имеющую очень широкую шину, что давало бы огромную пропускную способность. Но, увы, все не так просто. Чем шире шина, тем больше взаимных наводок создают друг для друга провода, по которым передаются данные, чем выше частота передачи данных, тем большие требования предъявляются к технологии изготовления, и тем выше потребляемая мощность.

Обычные типы памяти (FPM или SDRAM) иногда называют устройствами с широким каналом. Ширина канала памяти равна ширине шины данных процессора (в Pentium-системах - 64 бита). Память DR DRAM является устройством с узким каналом передачи данных - количество данных, передаваемых за один такт, достигает только 16 бит. Для повышения производительности i1084 можно использовать двухканальные DR DRAM (т.е. данные в них передаются от двух модулей - пока с одного модуля считываются данные, другой готовится к передаче, затем модули меняются ролями). Также было предложено еще одно конструктивное решение - передача управляющей информации отделена от передачи данных по шине.

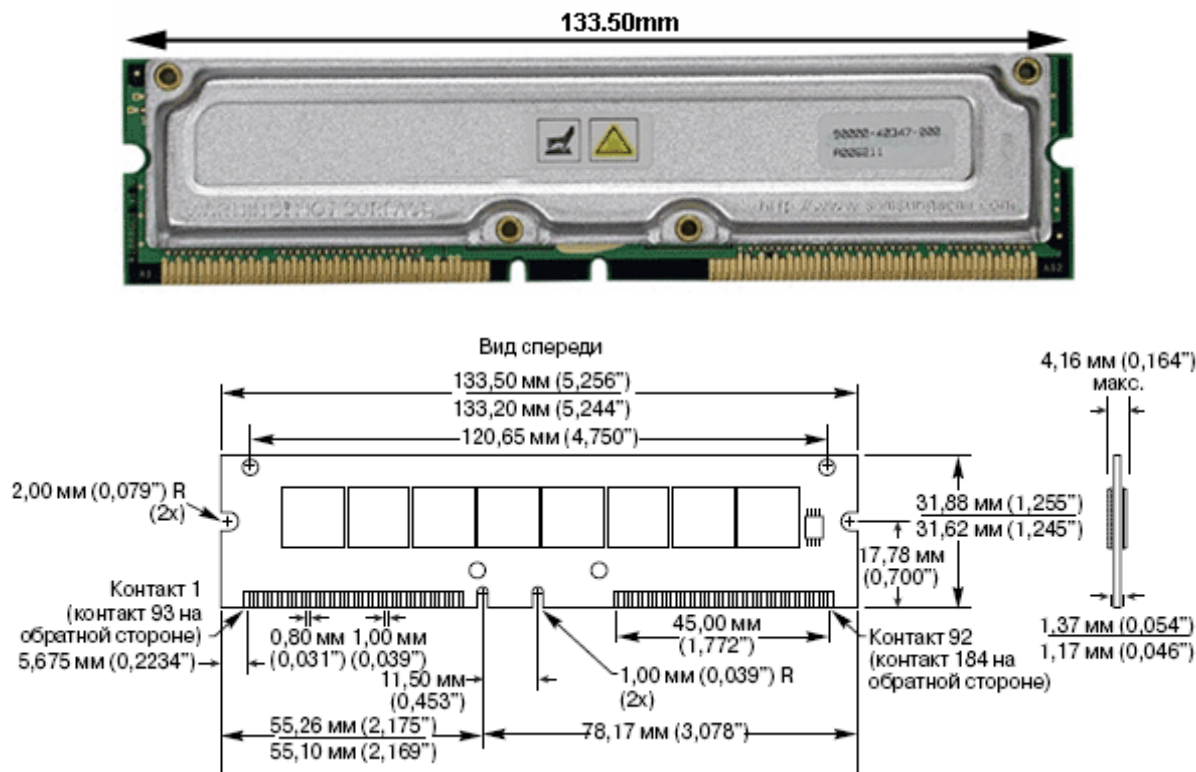
Итак, Rambus пошел по пути изготовления очень высокоскоростной шины (400, 533 МГц, сравните с 133 МГц SDRAM и 133, 166, 200 DDR SDRAM), но при этом очень узкой. Разрядность модулей RD DRAM - всего 16 бит, или 32 бита в двухканальных модулях

(сравните с 64 бит у SDRAM и DDR SDRAM). Итого, пропускная способность такого канала обмена составляет $800 \text{ МГц} * 2 \text{ байта} = 1600 \text{ Мбайт/с}$. Т.е. пропускная способность канала Rambus выше, чем у SDRAM (800 Мбайт/с при частоте 100 МГц и 1064 Мбайт/с при частоте 133 МГц), равна DDR SDRAM при частоте 100 МГц, и проигрывает DDR SDRAM на частоте 133 МГц ($133 \text{ МГц} * 64 \text{ бит} * 2 \text{ из-за DDR} = 2100 \text{ Мбайт/с}$).

Но при этом есть еще два немаловажных фактора. Первое: у RD DRAM задержки на доступ к памяти выше, чем у SDRAM и DDR SDRAM, что при равенстве пропускной способности уже ставит под сомнение эффективность применения RD DRAM. И второе: дело в том, что цена на DR DRAM выше, чем цена на DDR SDRAM.

Одно время память этого типа считалась очень перспективной - на нее делала ставку в своих системах фирма Intel, несмотря на дикую цену и совсем не оправдывающую себя скорость. И это было большой ошибкой Intel - сегодня RIMM-память применяется чаще всего в старых серверах и рабочих станциях, кроме того, существует очень мало чипсетов, и следовательно, материнских плат, поддерживающих память на микросхемах DR DRAM. Именно поэтому сейчас на рынке

памяти персональных компьютеров безоговорочно доминирует DDR SDRAM, на смену ей идее DDRII SDRAM, а применение Rambus осталось прежним – SGI и игровые приставки (например Sony Playstation II или III). Рассмотрим, какие модули Rambus применялись:



RIMM 184 pin. Применялись с 16-битным вариантом Rambus, имели 1 или 2 ключа (прорези) в центральной части модуля, положение и количество которых зависело от используемого напряжения питания. Маркировались такие модули частотой передачи данных, т.е. модули, работающие на частоте 400 МГц, передавали данные с частотой 800 МГц, и носили маркировку PC800.



RIMM 232 pin. Применяются с 32-битным вариантом Rambus, также имеют 1 или 2 ключа. Маркируются такие модули максимальной пропускной способностью, которую могут обеспечить, т.е. так же как и DDR SDRAM.

2.5. Графическая память

Как известно, оперативная память применяется для нужд не только центрального процессора, но и графического процессора. В современных графических видеокартах используется так называемая графическая память, микросхемы которой распаиваются на плате графической карты. Аналогично тому, что существуют различные типы оперативной памяти (SDR, DDR, DDR2 и DDR3), графическая память тоже бывает разной. Для того чтобы отличать оперативную память от графической памяти, последнюю снабжают обозначением «G». Так, бывает память GDDR2, GDDR3 и GDDR4. Несмотря на схожие названия (GDDR2 и DDR2, GDDR3 и DDR3), графическая память существенно отличается от оперативной памяти.

Отметим, что впервые графическая память GDDR2 (Graphics Double Data Rate, version 2) была использована компанией NVIDIA в видеокарте на базе процессора GeForce FX 5800. В то же время по принципу действия графическая память GDDR2 не имеет ничего общего с памятью

DDR2 и в этом смысле более схожа с памятью DDR. В частности, в памяти GDDR2 не используется технология $4n$ -Prefetch, когда буфер ввода-вывода данных работает на удвоенной частоте. От обычной DDR-памяти память GDDR2 отличается более высокими тактовыми частотами, требованиями к напряжению и способами терминирования сигналов.

Память GDDR3 (Graphics Double Data Rate, version 3) была разработана компанией ATI, однако впервые использовалась на видеокартах с графическим процессором NVIDIA GeForce FX 5700 Ultra.

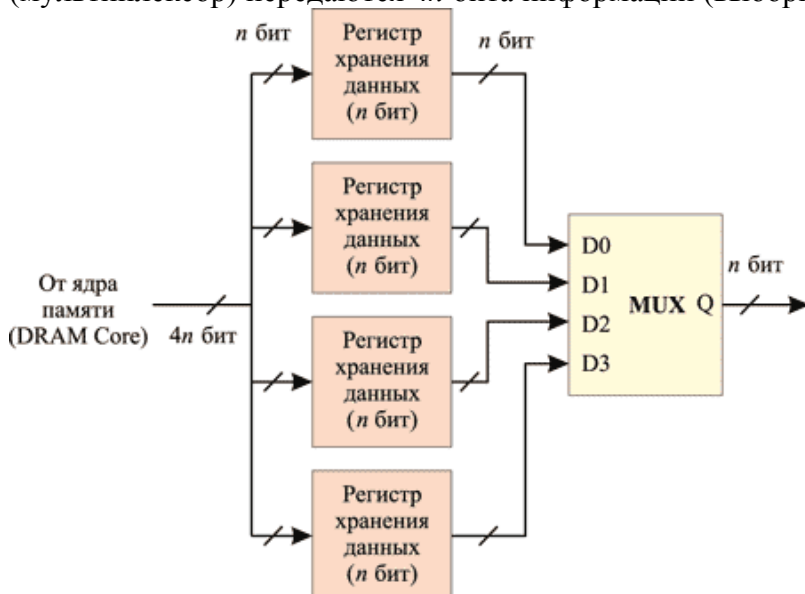
Эта память также не имеет никакого отношения к памяти DDR3 и по принципу действия более схожа с памятью DDR2, отличаясь от нее тактовыми частотами, требованиями к напряжению и способами терминирования сигналов. В памяти GDDR3, так же как и в памяти DDR2, используется технология $4n$ -Prefetch.

Память GDDR4 (Graphics Double Data Rate, version 4) сегодня применяется только в видеокарте с процессором ATI Radeon, анонсированном компанией 21 июля 2007 года. Хотя на сегодняшний момент компания NVIDIA заявила о поддержке GDDR4 в своей новой линейке видеокарт. Этот вид памяти является своеобразным аналогом памяти DDR3 в том смысле, что в ней реализован механизм $8n$ -Prefetch.

2.6. Принцип работы ОЗУ (дополнительная информация, не обязательно)

Теперь мы с вами уже знаем для чего предназначена Оперативная память, какие разновидности бывают. Осталось только определиться, как же она работает. И сделаем мы это с вами на примере памяти DDR2, как на наиболее распространенной на сегодняшний момент видом памяти.

Если следовать терминологии DDR (Double Data Rate), то память DDR2 было бы логично назвать не DDR2 а QDR (Quadra Data Rate), поскольку этот стандарт подразумевает в четыре раза большую скорость передачи, то есть в стандарте DDR2 при пакетном режиме доступа данные передаются четыре раза за один такт. Для организации данного режима работы памяти необходимо, чтобы буфер ввода-вывода (мультиплексор) работал на учетверенной частоте по сравнению с частотой ядра памяти. Достигается это следующим образом: ядро памяти, как и прежде, синхронизируется по положительному фронту тактирующих импульсов, а с приходом каждого положительного фронта по четырем независимым линиям в буфер ввода-вывода (мультиплексор) передаются $4n$ бита информации (выборка $4n$ битов за такт, $4n$ -Prefetch).

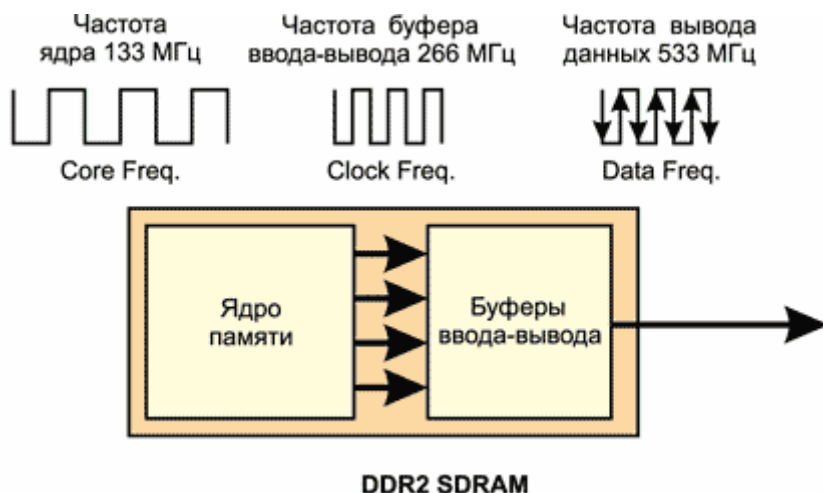


Сам буфер ввода-вывода тактируется на удвоенной частоте ядра памяти и синхронизируется как по положительному, так и по отрицательному фронту этой частоты. Иными словами, с приходом положительного и отрицательного фронтов происходит передача битов в мультиплексном режиме на шину данных (как изображено на рисунке). Это позволяет за каждый такт работы ядра памяти передавать четыре слова на шину данных, то есть вчетверо повысить пропускную способность памяти.

По сравнению с памятью DDR память DDR2 позволяет обеспечить ту же пропускную способность, но при вдвое меньшей частоте ядра. К примеру, в памяти DDR400 ядро функционирует на частоте 200 МГц, а в памяти DDR2-400 — на частоте 100 МГц. В этом смысле память DDR2 имеет значительно большие потенциальные возможности для увеличения пропускной способности по сравнению с памятью DDR.

В памяти DDR2 реализована схема разбиения массива памяти на четыре логических банка, а для модулей емкостью 1 и 2 Гбайт — на восемь логических банков.

Рассмотрим упрощенную схему работы DDR2-памяти на примере операции чтения. Пусть имеются два банка памяти (Bank0, Bank1), длина пакета (Burst Length) равна 4, $t_{CAS} = 2$ и $t_{RCD} = 3$, $t_{RRD} = 2$. Первоначально необходимо активировать оба банка и получить доступ к строке в этом банке. Тогда через каждые 2 такта активируется новый банк, а через каждые три такта после активации банка следует команда чтения данных из этого банка.

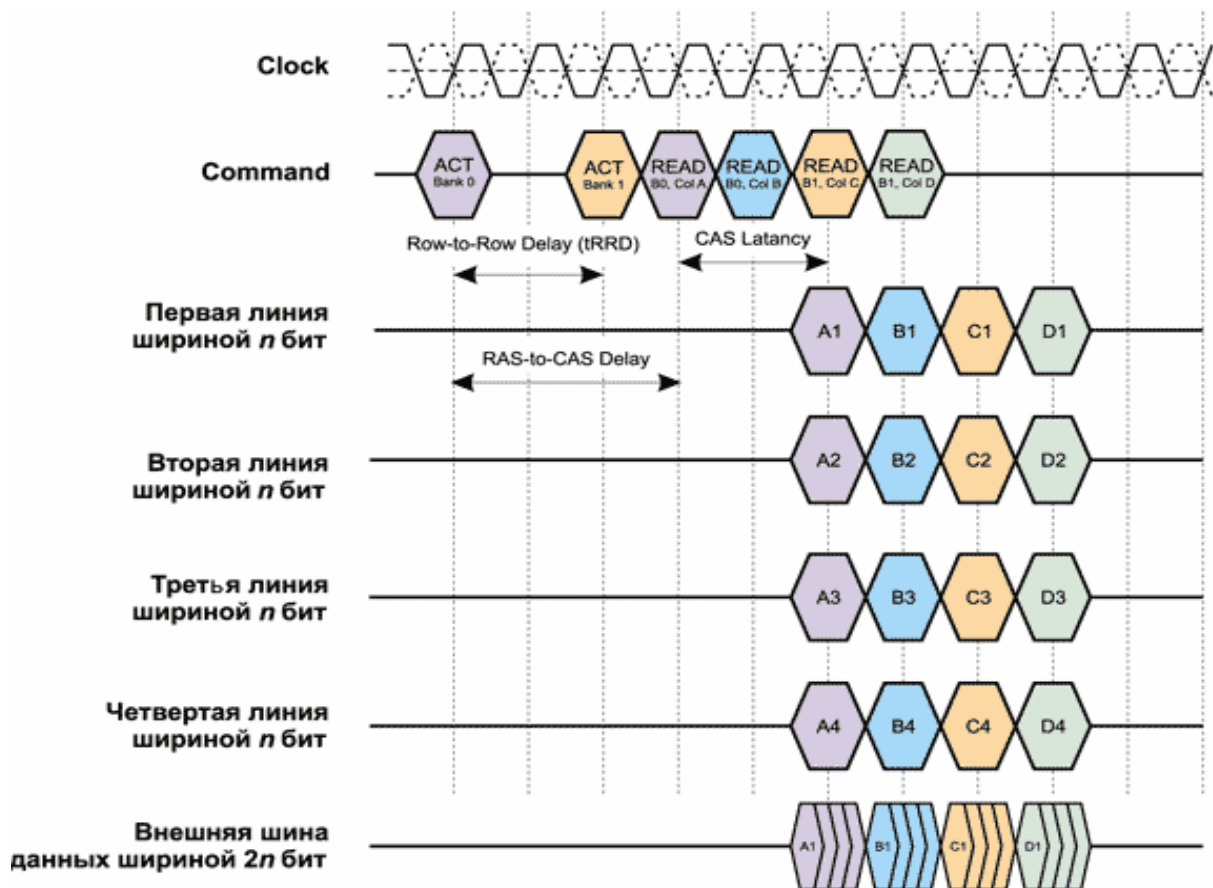


Поскольку задержка CAS Delay составляет два такта, то через два такта после команды чтения данные могут быть считаны с шины данных. Напомним, что у нас имеется четыре шины данных (линии) шириной n бит каждая и передача данных может происходить параллельно по каждой из этих линий. В нашем упрощенном примере можно считать, что слова A1-A4, соответствующие первому банку, одновременно (в течение одного такта) передаются по четырем линиям. На следующем такте по четырем линиям одновременно передаются слова B1-B4 и т.д.

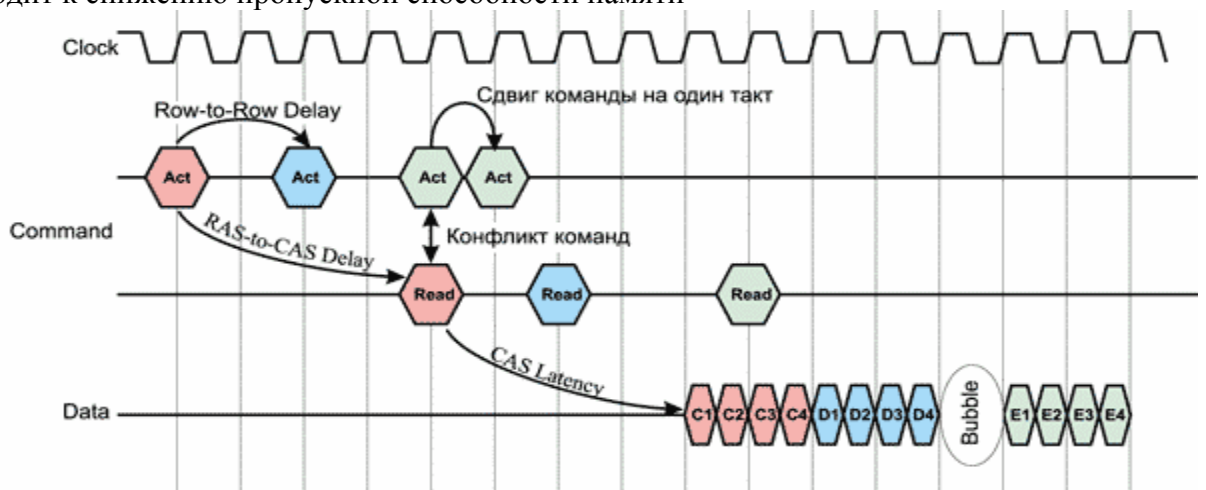
Далее эти данные передаются в мультиплексор синхронно с положительным фронтом тактового импульса. Поскольку мультиплексор работает на удвоенной частоте и выводит данные по шине шириной n бит синхронно с положительным и отрицательным фронтами, за один такт работы ядра памяти осуществляется вывод на шину данных $4n$ бит (4 слова).

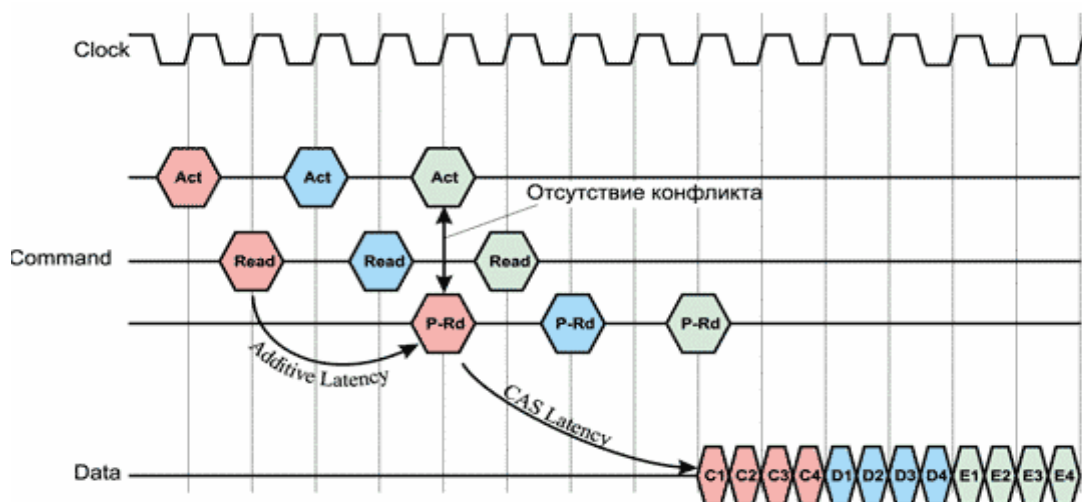
Понятно, что в случае реализации архитектуры $4n$ -Prefetch длина пакета (Burst Length) данных не может быть менее 4. Поэтому для памяти DDR2 минимальная длина пакета составляет 4.

Одна из главных задач в технологии $4n$ -Prefetch — обеспечить наличие непрерывного потока данных на каждой из четырех линий шириной n бит. С учетом того, что команды тактируются на частоте работы ядра памяти и в один момент времени на шине может присутствовать только одна команда, эта задача не такая простая, как кажется.



Рассмотрим в качестве гипотетического примера ситуацию с тремя банками памяти. Активация каждого следующего банка может происходить только после промежутка времени Row-to-Row Delay (tRRD). Типичным является случай, когда tRRD составляет два такта. Кроме того, для каждого отдельного банка после его активации команда на чтение (выбор столбца в пределах активированной строки) поступает с задержкой, определяемой RAS-to-CAS Delay (tRCD). И если $tRCD = 4T$, то команда на чтение первого банка совпадет с активацией третьего банка. Для того чтобы избежать конфликта команд, команду активации третьего банка приходится смещать на целый цикл, что, естественно, приводит и к смещению всех последующих команд для этого банка. В результате такого сдвига на шине данных образуется пропуск или пузырь (Bubble), что приводит к снижению пропускной способности памяти.





2.7. Логическая организация памяти (дополнительная информация, не обязательно)

Мы с вами говорили про то как работает память, хотелось бы немного упомянуть про то как она устроена.

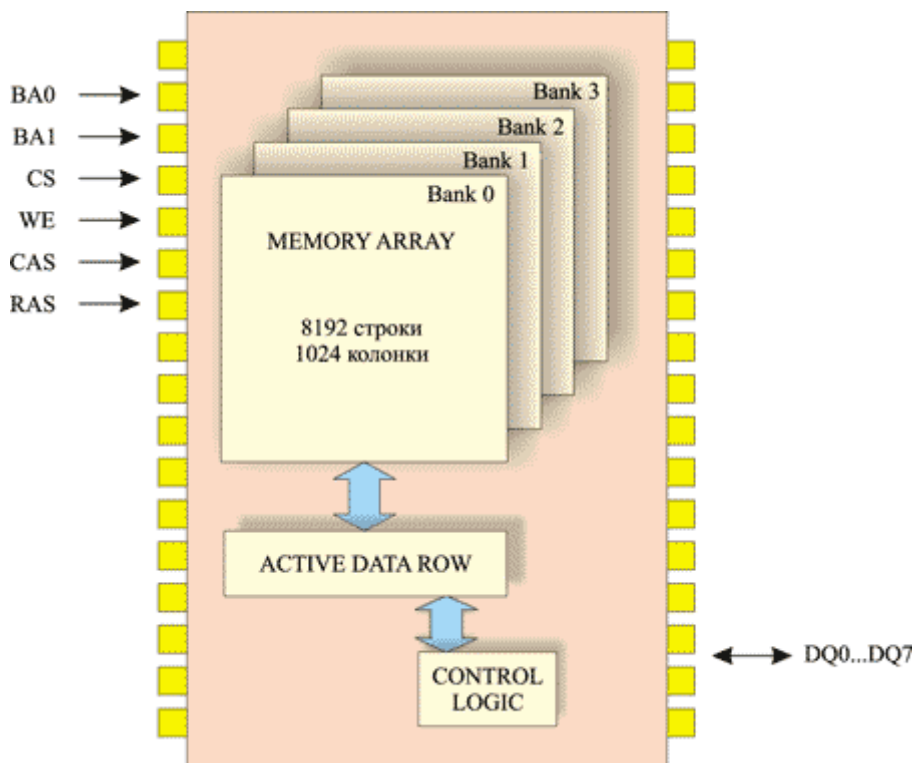


Рисунок 5.1 - Упрощенная схема чипа памяти с шириной шины данных 8 бит

Микросхемы памяти организованы в виде матрицы, напоминающей лист бумаги в клетку, причем пересечение столбца и строки матрицы создает одну из элементарных ячеек. Понятно, что количество строк и столбцов в матрице памяти определяет ее объем в битах или байтах. Кроме того, современные чипы памяти имеют несколько банков, каждый из которых можно рассматривать как отдельную матрицу со своими столбцами и строками.

На рисунке показана упрощенная схема чипа памяти, в котором имеется четыре банка, каждый из которых содержит 8192 строки и 1024 столбца. Таким образом, емкость каждого банка составляет: $8192 \times 1024 = 8192 \text{ Кбайт} = 8 \text{ Мбайт}$.

С учетом того, что чип содержит четыре банка, получается, что полная емкость чипа равна 32 Мбайт.

Кроме емкости и количества банков, каждая микросхема (чип) памяти характеризуется шириной шины данных, то есть количеством линий для передачи данных (DQ0, DQ1 и т.д.). Подчеркнем, что здесь речь идет не о модуле памяти, а об отдельном чипе памяти, из которых состоят модули. Ширина шины данных современных чипов памяти может составлять 4, 8, 16 или 32 линии, и именно этот параметр определяет логическую структуру чипа памяти. К примеру, чип емкостью 1024 Мбит при ширине шины данных 8 бит может быть составлен как 128 М x 8, то есть может рассматриваться и как 128 М логических 8-битных элементов. Тот же чип 1024 Мбит можно составить и как 64 М x 16 или 32 М x 32. В записи 64 М x 16, определяющей логическую структуру чипа памяти, первая цифра называется глубиной чипа памяти, а вторая — шириной.

Если говорить не об отдельном чипе памяти, а о модуле памяти, который включает несколько чипов памяти, то, кроме емкости, выражаемой в мегабайтах (Мбайт) или в гигабайтах (Гбайт), он характеризуется шириной или разрядностью интерфейса шины данных. Для всей современной памяти ширина шины данных составляет 64 бита. Ширина шины данных модуля памяти (64 бит) больше, чем ширина шины данных отдельного чипа памяти (4, 8, 16 или 32 бит). Для того чтобы согласовать передачу данных, используется простое слияние шин данных отдельных чипов памяти в шину данных модуля памяти.

Такое «заполнение» шины данных памяти принято называть составлением физического банка памяти. К примеру, для составления одного физического банка 64-разрядного модуля памяти SDRAM необходимо наличие 16 чипов шириной x4, или 8 чипов шириной x8, или 4 чипов шириной x16.

Как и отдельный чип памяти, модуль памяти характеризуется глубиной, которая определяется как емкость модуля, выраженная в битах, деленная на разрядность шины данных (64 бит). Соответственно произведение ширины на глубину дает полную емкость модуля и определяет его организацию. Так, для модуля памяти емкостью 1024 Мбайт справедлива запись 128 М x 64.

Но все бы было хорошо если бы так просто. По своей логической структуре память разделяется на области или зоны, это:

- **Conventional memory** – основная память;
- **UMA (Upper Memory Area)** – верхняя память;
- **HMA (High Memory Area)** – область верхней памяти или область верхних адресов;
- **XMS (eXtended Memory Specification)** – дополнительная память;
- **EMS (Expanded Memory Specification)** – расширенная память;

Спрашивается зачем все так сложно, и зачем было память делить на области.

Тянется это еще с компьютеров PC/XT, когда рабочее пространство оперативной памяти составляло 1 Мбайт. Это пространство было разделено на области, в одну область грузились специальные программы DOS, в другую – все остальное. Деление, по началу было пополам: по 512 Кбайт для каждой области. Затем было решено, что для обслуживания системы вполне хватит 384 Кбайт. Это означало, что для всего остального оставалось уже 640 Кбайт. Сегодня объем оперативной памяти шагнул за сотни мегабайт, но деление осталось и еще более усложнилось.

Что бы рассказывать дальше о зонах памяти, необходимо сделать большое отступление. Тот, кто разбирается в адресации, может этот абзац не читать, потому как сейчас пойдет разговор именно о ней. Если Вам кто-либо сказал, что оперативная память и адресное пространство одно и то же – неверьте! Адресное пространство – это набор адресов, который может формировать процессор. Зачем? Хороший вопрос. Дело в том, что каждая ячейка памяти имеет адрес. И что бы считать (или записать) хранимую в ней информацию, надо к ней обратиться по ее адресу. Адреса делятся на виртуальные (логические) и физические. Физические адреса – это реальные адреса реальных ячеек памяти. Программам глубоко параллельно до таких адресов, так как они оперируют символьными именами, которые затем транслятором преобразовываются в виртуальные адреса. Потом виртуальные адреса преобразовываются в физические. Делается это аппаратным способом, а вот как - я сейчас рассказывать не буду, потому что это тема отдельного разговора. Логические адреса представляются в шестнадцатеричной форме и состоят из двух частей. Почему из двух? Дело в том, что логически оперативная память разделена на сегменты. Так вот первая часть логического адреса – начало сегмента, а вторая – смещение от этого начала (сегмент:смещение). Выглядит это примерно так: D000:7FFF. Кстати, записать подобный адрес

можно и по-другому, сложив обе части. Получим D7FFF – это будет полный, а точнее линейный адрес (в шестнадцатеричной системе счисления принято еще букву h в конце адреса добавлять, но мы же с Вами об этом уже знаем ;) и я этого делать не буду). Объем адресуемой памяти не безграничен. Он зависит от адресной шины процессора, а точнее от ее разрядности. Давайте-ка посчитаем. В процессоре 8086 использовалась 20 разрядная адресная шина. Так как в компьютерах используется двоичная система, то возведя 2 в степень 20, получим максимальный адресуемый объем памяти для 8086 процессора. Это будет 1 048 576 байт или 1 Мбайт. Для современных процессоров максимальный адресуемый объем памяти равен 64 Гбайт. Нам и надо было только себе уяснить, что из себя представляют адреса. Двигаемся дальше!

Основная память (Conventional memory)

Начинается с адреса 00000 (0000:0000) и до 90000 (9000:0000). Это занимает 640 Кбайт. В эту область грузится в первую очередь таблица векторов прерываний, начиная с 00000 и занимает 1 Кбайт, далее следуют данные из BIOS (счетчик таймера, буфер клавиатуры и т. д.), а затем уж всякие 16 разрядные программы DOS (для них 640 Кбайт – барьер, за который могут выскочить только 32 разрядные проги). На данные BIOS'a отводится 768 байт. С этой закончили :).

Верхняя память (UMA)

Начинается с адреса A0000 и до FFFFF. Занимает она 384 Кбайт. Сюда грузится инфа, связанная с аппаратной частью компьютера. UMA можно разделить на 3 части по 128 Кбайт. Первая часть (от A0000 до BFFFF) предназначена для видеопамати. В следующую часть (от C0000 до DFFFF) грузятся программы BIOS адаптеров. Последняя часть (от E0000 до FFFFF) зарезервирована для системной BIOS. Тут есть одна фишка. Дело в том, что последние 128 Кбайт не полностью используются. В большинстве случаев под BIOS задействованы только последние 64 Кбайт. Свободная же часть UMB управляется драйвером EMM386.EXE и используется для нужд операционной системы.

Дополнительная память (XMS)

Основная и верхняя память занимают 1 Мбайт памяти в общей сложности. Что бы работать с областью свыше 1 Мбайта, процессор должен работать в защищенном режиме. Эта область называется дополнительная память (XMS). Что бы работать в XMS используя DOS, для процессоров был разработан еще один режим – виртуальный. Помните, еще в начале раздела мы говорил, что DOS не может переплюнуть барьер в 640 Кбайт? Так вот, виртуальный режим позволяет разбить дополнительную память на части по 1 Мбайту. В каждую часть грузится по программе DOS и там они варятся в реальном режиме но уже не мешая друг другу выполняться одновременно. 32 разрядным приложениям барьер в 640 Кбайт безразличен и для них деление XMS ни к чему. Отвечает за перевод режимов процессора драйвер EMM386.EXE, а за организацию самой области – HIMEM.SYS. Посмотреть, что творится у Вас в XMS можно с помощью SysInfo из набора Norton Utilities.

Область верхней памяти или область верхних адресов (HMA)

Но это еще не все. В дополнительной области, в самом начале ее первого мегабайта выделена зона, объем которой равен 64 Кбайт минус 16 байт. Называется это областью верхних адресов (HMA). История появления этой области лежит корнями глубоко и тянется аж к 80286 процессору, а точнее к ошибке в его схеме. Я уже говорил, что процессоры 8086 и 8087 имели 20 разрядную адресную шину, работали в реальном режиме и могли максимально обратиться по адресу FFFFF (FFFF:000F). А вот 80286 процессор имел уже 24 разрядную шину адреса, работал в реальном и защищенном режимах и мог адресовать до 16 Мбайт памяти. Теперь рассмотрим такую вот ситуацию: возьмем сегментный адрес FFFF:FFFF и переведем его в линейный, получим 10FFEF. Такой адрес 8086 процессор адресовать не мог, так как это уже во втором мегабайте памяти. В подобных случаях делалось просто – откидывался старший разряд. Получится 0FFEF, а это уже обращение на 16 байт от конца первого сегмента в 64 Кбайт первого мегабайта памяти. И все отлично. 80286 процессор работая в реальном режиме должен был поступать также. Но проблема заключалась в том, что в этом режиме 21 линия шины адреса (A20) не отключалась и оставалась в работе. А значит получалось в данном случае обращение на 16 байт от конца первого

сегмента в 64 Кбайт второго мегабайта памяти. Не все приложения могли тогда работать таким образом и инженеры нашли способ включать и отключать линию A20. Для этих целей использовался специальный контроллер. Для управления НМА используется уже известный нам HIMEM.SYS.

Расширенная память (EMS)

Ну и наконец еще одна область – расширенная память (EMS). На сегодняшний день этот вид памяти мало кому нужна и мало кем используется. Поэтому я только лишь коротенько пробежусь по данной теме. Находится эта область в верхней памяти и занимает порядка 64 Кбайт. Использовалась она лишь в старых компьютерах с оперативной памятью до 1 Мбайта. В силу своей спецификации это достаточно медленная область. Дело в том, что расширенная память – это один из многих коммутируемых сегментов. После того, как сегмент заполнится, происходит смена использованного сегмента новым. Но работать можно только с одним сегментом, а это, Вы сами должны понимать, не совсем хорошо, удобно и быстро. Как правило первый сегмент EMS находится по адресу D000.

Если все выше изложенное изобразить в графическом виде получится:

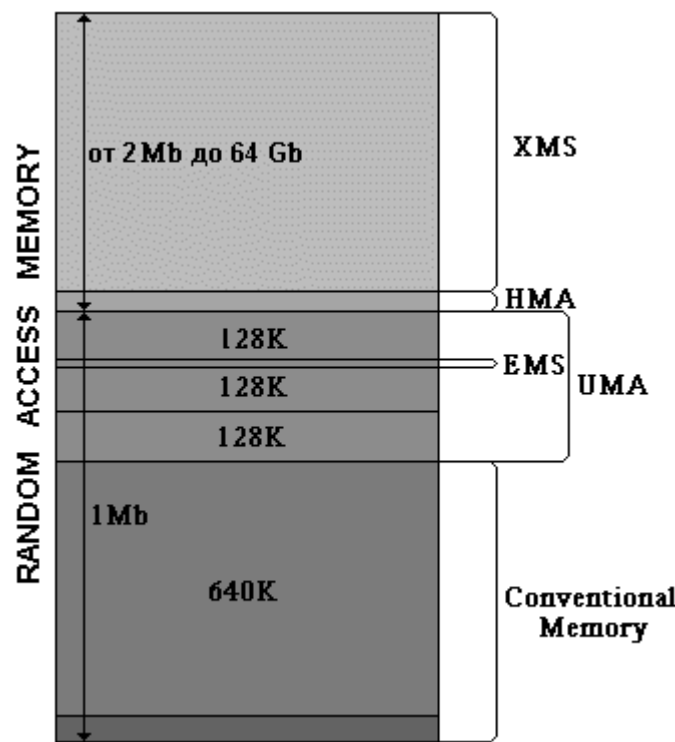


Рисунок 5.2 – Графическое отображение, Логической организации памяти.

2.8. Таблицы характеристик современных видов памяти

Таблица соответствия названий, пропускной способности и длительности такта для Оперативной памяти DDR SDRAM

Standard name	Memory clock	Cycle time	I/O Bus clock	Data transfers per second	Module name	Peak transfer rate
DDR-200	100 MHz	10 ns	100 MHz	200 Million	PC-1600	1600 MB/s
DDR-266	133 MHz	7.5 ns	133 MHz	266 Million	PC-2100	2100 MB/s
DDR-300	150 MHz	6.67 ns	150 MHz	300 Million	PC-2400	2400 MB/s
DDR-333	166 MHz	6 ns	166 MHz	333 Million	PC-2700	2700 MB/s
DDR-400	200 MHz	5 ns	200 MHz	400 Million	PC-3200	3200 MB/s

DDR-500	250 MHz	4 ns	250 MHz	500 Million	PC-4000	4000 MB/s
*DDR-650	325 MHz	3.1 ns	325 MHz	650 Million	PC-5200	5200 MB/s

Таблица соответствия названий, пропускной способности, и длительности такта для Оперативной памяти DDR2 SDRAM

Standard name	Memory clock	Cycle time	I/O Bus clock	Data transfers per second	Module name	Peak transfer rate
DDR2-400	100 MHz	10 ns	200 MHz	400 Million	PC2-3200	3200 MB/s
DDR2-533	133 MHz	7.5 ns	266 MHz	533 Million	PC2-4200	4266 MB/s
					PC2-4300	
DDR2-667	166 MHz	6 ns	333 MHz	667 Million	PC2-5300	5333 MB/s
					PC2-5400	
DDR2-800	200 MHz	5 ns	400 MHz	800 Million	PC2-6400	6400 MB/s
DDR2-1066	266 MHz	3.75 ns	533 MHz	1066 Million	PC2-8500	8533 MB/s

Таблица соответствия названий, пропускной способности, и длительности такта для Оперативной памяти DDR3 SDRAM

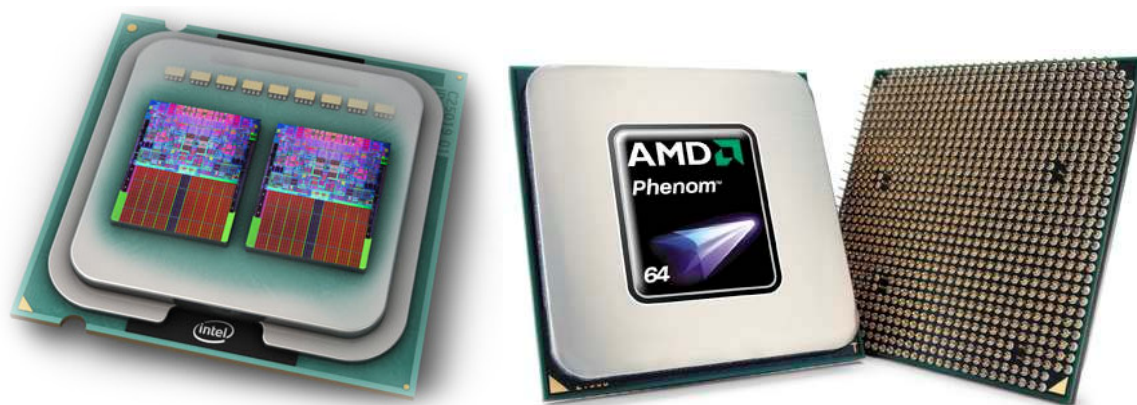
Standard name	Memory clock	Cycle time	I/O Bus clock	Data transfers per second	Module name	Peak transfer rate
DDR3-800	100 MHz	10 ns	400 MHz	800 Million	PC3-6400	6400 MB/s
DDR3-1066	133 MHz	7.5 ns	533 MHz	1066 Million	PC3-8500	8533 MB/s
DDR3-1333	166 MHz	6 ns	667 MHz	1333 Million	PC3-10600	10667 MB/s
DDR3-1600	200 MHz	5 ns	800 MHz	1600 Million	PC3-12800	12800 MB/s
DDR3-1800	225 MHz	4.4 ns	900 MHz	1800 Million	PC3-14400	14400 MB/s
DDR3-2000	250 MHz	4 ns	1000 MHz	2000 Million	PC3-16000	16000 MB/s
DDR3-2600	325 MHz	3.1 ns	1300 MHz	2600 Million	PC3-20800	20800 MB/s

3. Процессоры.

3.1. Назначения процессора.

Для начала давайте определимся, что такое процессора и для чего он нам необходим.

Центральный процессор (сокращенно — ЦП, или микропроцессор, или просто препроцессор) стал одной из важнейших разработок технологии интегральных микросхем за все время ее развития (рисунок 1.1). На первый взгляд центральный процессор является довольно неинтересным устройством — несмотря на свою чрезвычайную сложность, типичный процессор выполняет всего лишь три типа функций: математические вычисления, логические сравнения и обработка данных. Это не слишком большой репертуар для устройства, состоящего из нескольких сотен миллионов транзисторов. Однако главной особенностью этого устройство является не количество выполняемых им функций, а то, что каждая из этих функций является частью программы, которую процессор может читать и выполнять. Изменяя программу, можно заставить процессор выполнять различную работу без необходимости внесения изменений в физические компоненты компьютера.



После рождения концепции центрального процессора разработчики поняли, что одно и то же устройство можно использовать для решения чрезвычайно широкого круга задач, задавая ему нужный набор команд. Эта идея привела к созданию современного компьютера, выделив в виде независимых областей создание аппаратного и программного обеспечения.

То есть подведем итог, получается что процессор является основным вычислительным блоком компьютера, с наибольшей степенью определяющим его мощь.

3.2 Архитектуры процессоров

Сейчас существует множество процессорных архитектур, которые делятся на три глобальные категории – RISC, CISC и MISC

RISC – Reduced (Restricted) Instruction Set Computer – процессор с сокращенной системой команд. Эти процессоры обычно имеют набор однородных регистров универсального назначения, причем их число может быть большим. Система команд отличается простотой, коды инструкций имеют четкую структуру, как правило, с фиксированной длиной. В результате аппаратной реализации такой внутренней архитектуры, позволяющей с небольшими затратами декодировать и выполнять эти инструкции за минимальное (в пределе 1) число тактов синхронизации. Определенное преимущества дает и унификация регистров.

CISC – Complete Instruction Set Computer – процессоры с полным набором инструкций, к которым относятся и семейство x86. Состав и назначение их регистров существенно неоднородны,

широкий набор команд усложняет декодирование инструкций, на что расходуются аппаратные ресурсы. Возрастает число тактов, необходимых для выполнения инструкций.

Для большего понимания давайте рассмотрим на примере установки электрической лампочки, инструкции CISC.

- 1. Возьмите электрическую лампочку.*
- 2. Вставьте ее в патрон.*
- 3. Вращайте до отказа.*

И аналогичный пример в виде инструкций RISC.

- 1. Поднесите руку к лампочке.*
- 2. Возьмите лампочку.*
- 3. Поднимите руку к патрону.*
- 4. Вставьте лампочку в патрон.*
- 5. Поверните ее.*
- 6. Лампочка поворачивается в патроне? Если да, то перейти к п. 5.*
- 7. Конец.*

Многие инструкции RISC довольно просты, поэтому для выполнения какой либо операции потребуется больше таких инструкций. Их основное преимущество состоит в том, что процессор осуществляет меньше операций, а это, как правило, сокращает время выполнения отдельных команд и соответственно всей задачи (программы). Можно долго спорить о том, что же в действительности лучше — RISC или CISC, хотя, по правде говоря, такого понятия, как “чистая” микросхема RISC или CISC, не существует. Подобная классификация не более чем вопрос терминологии.

MISC-Minimum Instruction Set Computing — 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).

Процессоры, образующие «компьютеры с минимальным набором команд» MISC, как и процессоры RISC, характеризуются небольшим числом чаще всего встречающихся команд. Вместе с этим, принцип «очень длинных слов команд» VLIW (Very long instruction word) обеспечивает выполнение группы непротиворечивых команд за один цикл работы процессора. Порядок выполнения команд распределяется таким образом, чтобы в максимальной степени загрузить маршруты, по которым проходят потоки данных. Таким образом архитектура MISC объединила вместе суперскалярную и VLIW концепции. Компоненты процессора просты и работают с высокими скоростями.

Стоит заметить, что под архитектурой процессора, мы понимаем его программную модель, то есть программно видимые свойства самого процессора.

3.2.1 Суперскалярная архитектура

Суперскалярная архитектура - Способность выполнения нескольких машинных инструкций за один такт процессора. Появление этой технологии привело к существенному увеличению производительности. А теперь поговорим об этом поподробнее.

Основная черта всех современных процессоров состоит в том, что они способны запускать на исполнение не только ту команду, которую (согласно коду программы) следует исполнить в данный момент времени, но и другие, следующие после неё. Приведём простой (канонический) пример. Пусть нам следует исполнить следующую последовательность команд:

- (1) $A = B + C$
- (2) $Z = X + Y$
- (3) $K = A + Z$

Легко заметить, что команды (1) и (2) совершенно независимы друг от друга — они не пересекаются ни по исходным данным (переменные B и C в первом случае, X и Y во втором), ни по месту размещения результата (переменная A в первом случае и Z во втором). Стало быть, если на данный момент у нас есть свободные исполняющие блоки в количестве более одного, данные команды можно распределить по ним, и выполнить одновременно, а не последовательно. Таким образом, если принять время исполнения каждой команды равным N тактов процессора, то в классическом случае исполнение всей последовательности заняло бы $N \cdot 3$ тактов, а в случае с параллельным исполнением — всего $N \cdot 2$ тактов (так как команду (3) нельзя выполнить, не дождавшись результата исполнения двух предыдущих).

Замечание: разумеется, степень параллелизма не бесконечна: команды могут быть выполнены параллельно только в том случае, когда на данный момент времени есть в наличии соответствующее количество свободных от работы блоков функционального устройства (ФУ), причём именно таких, которые «понимают» рассматриваемые команды. Самый простой пример: блок, относящийся к ALU , физически неспособен исполнить инструкцию, предназначенную для FPU . Обратное также верно.

На самом деле всё ещё сложнее. Так, если у нас имеется следующая последовательность:

- (1) $A = B + C$
- (2) $K = A + M$
- (3) $Z = X + Y$

То очередь исполнения команд процессором будет изменена! Так как команды (1) и (3) независимы друг от друга (ни по исходным данным, ни по месту размещения результата), они могут быть выполнены параллельно — и будут выполнены параллельно. А вот команда (2) будет выполнена после них (третьей) — поскольку для того, чтобы результат вычислений был корректен, необходимо, чтобы перед этим была выполнена команда (1). Именно поэтому обсуждаемый в данном разделе механизм и называется «внеочередным исполнением команд» (**Out-of-Order Execution**, или сокращённо «**OoO**»): в тех случаях, когда очерёдность выполнения никак не может сказаться на результате, команды отправляются на исполнение не в той последовательности, в которой они располагаются в коде программы, а в той, которая позволяет достичь максимального быстродействия.

Теперь вам должно стать окончательно понятно, зачем современным CPU такое количество однотипных исполняющих блоков: они обеспечивают возможность параллельного выполнения нескольких команд, которые в случае с «классическим» подходом к проектированию процессора пришлось бы выполнять в той последовательности, в которой они содержатся в исходном коде, одну за другой.

Процессоры, оснащённые механизмом параллельного исполнения нескольких подряд идущих команд, принято называть «суперскалярными». Однако не все суперскалярные процессоры поддерживают внеочередное исполнение. Так, в первом примере нам достаточно «простой суперскалярности» (выполнения двух последовательных команд одновременно) — а вот во втором примере без перестановки команд местами уже не обойтись, если мы хотим получить максимальное быстродействие. Все современные CPU обладают обоими качествами: являются суперскалярными, и поддерживают внеочередное исполнение команд. В то же время, были в истории x86 и «простые суперскаляры», OoO не поддерживающие.

Например, классическим десктопным x86-суперскаляром без OoO был Intel Pentium [MMX]. Справедливости ради, стоит заметить, что никаких заслуг в разработке концепций суперскалярности и OoO — нет ни у Intel, ни у AMD, ни у какого-либо иного (в том числе из ныне почивших) производителя x86 CPU. Первый суперскалярный компьютер, поддерживающий OoO, был разработан Сеймуром Креем (Seymour Cray) ещё в 60-х годах XX века. Для сравнения: Intel свой первый суперскалярный процессор (Pentium) выпустила в 1993 году, первый суперскаляр с OoO (Pentium Pro) — в 1995 году; первый суперскаляр с OoO от AMD (K5) увидел свет в 1996 году. Комментарии, как говорится, излишни...



Рисунок - Упрощенная модель работы конвейера

3.2.2 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, дешифрация команды, адресация операнда в ОЗУ, выборка операнда из ОЗУ, выполнение команды, запись результата в ОЗУ. Каждую из этих операций сопоставляют одной ступени конвейера.

Давайте посмотрим как это происходит. Современные процессоры обрабатывают данные конвейерным способом, т.е. одновременно в процессоре на различных стадиях происходит выполнение нескольких команд - одна команда практически близка к исполнению, другая - в середине процесса исполнения, третья только входит на конвейер и т.д., на рисунке пример первого из применяемых в PC конвейера. Таким образом удастся значительно повысить производительность, так как все части процессора одновременно задействуются, и каждая часть процессора выполняет свой маленький объем работы. При этом вводят понятие "количество стадий конвейера" - число элементарных задач на которые разбивается выполнение команды – на рисунке выделена каждая стадия. При этом если стадий мало, то недостаточно полно используются все части процессора одновременно (страдает производительность), и невозможно использовать большие тактовые частоты. А если стадий много, то появляется возможность использовать более высокие частоты процессора, но при этом увеличиваются задержки в процессе исполнения команд (тоже страдает производительность). Таким образом, производители процессоров выбирают некоторое оптимальное количество стадий конвейера, при котором процессор показывает наивысшую производительность. Количество стадий в конвейерах современных процессоров от 15 до 30, при этом производители столкнулись с ещё одной проблемой – хоть частоты процессоров одних производителей благодаря длинным конвейерам (30 стадий) выросли более 3-х гигагерц, но их производительность сравнима с процессорами конкурентов, у которых конвейер 15 стадий, и частоты намного меньше. Также, при работе конвейера возникает проблема, связанная с исполнением команд ветвления (условных переходов): если в процессор поступила команда ветвления, то ответ на вопрос о том, какие следующие команды должны быть исполнены будет известен ТОЛЬКО после того, как исполнится команда ветвления. Но пока она не исполнена, на конвейер могли бы поступить другие команды, которые бы постепенно исполнялись! Получается, что пока не будет выполнена команда ветвления, конвейер должен простаивать! И чем длиннее конвейер, тем дольше он простаивает.

Кроме того существуют и дополнительные факторы снижающие эффективность конвейера:

- *простой конвейера*, когда некоторые ступени не используются (напр., адресация и выборка операнда из ОЗУ не нужны, если команда работает с регистрами);

- **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*);
- **очистка конвейера:** при попадании в него команды перехода (эту проблему удаётся сгладить, используя предсказание переходов)

Решением этой проблемы стало **динамическое исполнение** команд. **Динамическое исполнение** - совокупность трех методов обработки информации в процессоре: **Предсказание ветвлений, анализ потока команд, упреждающее выполнение.** Динамическое выполнение - важнейшее архитектурное преимущество современных процессоров, оказывает серьезное влияние на производительность, но их мы рассмотрим чуть позже.

3.2.3 Архитектура NetBurst

Данная архитектур появилась совершенно недавно, и основной ее задачей было увеличение производительности современных двух и четырех ядерных процессоров компании Intel, за счет оптимизации процесса обработки команд:

- **Intel Wide Dynamic Execution** — технология выполнения большего количества команд за каждый такт, повышающая эффективность выполнения приложений и сокращающая энергопотребление. Каждое ядро процессора может выполнять до четырех инструкций одновременно с помощью 14-стадийного конвейера
- **Intel Intelligent Power Capability** — технология, с помощью которой для исполнения задач активируется работа отдельных узлов чипа по мере необходимости, что значительно снижает энергопотребление системы в целом
- **Intel Advanced Smart Cache** — технология использования общей для всех ядер кэш-памяти L2, что снижает общее энергопотребление и повышает производительность, при этом, по мере необходимости, одно из ядер процессора может использовать весь объем кэш-памяти при динамическом отключении другого ядра
- **Intel Smart Memory Access** — технология оптимизации работы подсистемы памяти, сокращающая время отклика и повышающая пропускную способность подсистемы памяти
- **Intel Advanced Digital Media Boost** — технология обработки 128-разрядных команд SSE, SSE2 и SSE3, (широко используемых в мультимедийных и графических приложениях) за один такт

3.2.4 Параллельная архитектура

Классическая Архитектура процессора обладает тем недостатком, что она последовательная. Какой бы огромный массив данных ни требовалось обработать, каждый его байт должен будет пройти через центральный процессор, даже если над всеми байтами требуется провести одну и ту же операцию. Этот эффект называется узким горлышком фон Неймана.

Для преодоления этого недостатка предлагались и предлагаются архитектуры процессоров, которые называются параллельными. Параллельные процессоры используются в суперкомпьютерах.

Возможными вариантами параллельной архитектуры могут служить (по классификации Флинна):

- SISD* — один поток команд, один поток данных;
- SIMD* — один поток команд, много потоков данных;
- MISD* — много потоков команд, один поток данных;
- MIMD* — много потоков команд, много потоков данных.

3.3. Элементы архитектуры процессора.

3.3.1. Тактовая частота

Быстродействие процессора во многом зависит от тактовой частоты, обычно измеряемое в мегагерцах (МГц). Тактовая частота определяется параметрами кварцевого резонатора, представляющего собой кристалл кварца в оловянной оболочке. Под воздействием электрического напряжения в кристалле кварца возникают колебания электрического тока с частой, определяемой формой и размерами кристалла. Частота этого переменного тока и называется тактовой частотой. Наименьшей единицей времени для процессора, как для логического устройства является период тактовой частоты или просто такт. На каждую операцию (выполнение команды) процессор затрачивает некоторое количество тактов.

Естественно, чем выше тактовая частота процессора, тем производительнее он работает, так как в единицу времени происходит большее количество тактов и выполняется большее количество команд. Естественно, более новые процессоры работают на все более высоких тактовых частотах (это достигается, в частности, улучшением технологии их изготовления) показывают большую производительность. Но тактовая частота не единственный фактор, определяющий производительность процессора. Ведь количество тактов, затрачиваемое на выполнение команд тоже можно менять. И если первые x86 процессоры на выполнение одной команды тратили в среднем около 12 тактов, в 286 и 386 этот показатель в среднем составлял около 4,5 тактов, в 486 - около 2 тактов, то в процессорах Pentium в среднем выполняется одна команда за такт. Ну а процессоры семейства Pentium 4 могут выполнять несколько команд одновременно (за счет параллельного исполнения команд). Различное количество тактов, затрачиваемое процессорами на выполнение команд, затрудняет их сравнение с использованием только лишь тактовой частоты. Гораздо удобнее использовать для измерения производительности среднее количество операций, выполняемое за один такт работы процессора. Есть единица измерения этого количества – MIPS – million integer per second, миллионы целочисленных операций в секунду, и MFLOPS - million floating per second, миллионы вещественных операций в секунду. Этими величинами измеряется производительность разных блоков процессора – ALU и FPU. ALU – арифметико-логическое устройство, выполняет обработку целых чисел, это основной вычислительный блок процессора, именно он определяет используемый основной набор команд, напомним, что для рассматриваемых нами процессоров – это x86. А FPU – floatingpoint unit, обрабатывает вещественные числа (с плавающей запятой), это дополнительный блок, который также называют математическим сопроцессором, о нём мы поговорим позже.

Как вы уже наверное догадались теоретически сравнивать два процессора нужно рассматривая быстродействие и тактовую частоту работы в совокупности: чем меньше тактов затрачивает в среднем процессор на исполнение команды, тем выше его эффективность (производительность) даже при неизменной тактовой частоте.

Например:

486 процессор (в среднем 2 такта на команду) на частоте 133 МГц работает даже медленнее, чем процессор Pentium (в среднем 1 такт на команду) на частоте 75МГц. Оценивать реальную производительность процессора в сравнении с другими весьма непросто, и нужно понимать, что такое сравнение во многом зависит от той задачи, которую процессор решает. Тактовые частоты современных процессоров лежат в диапазоне от 1,5 ГГц до 3,8 ГГц, и более.

Перечисленные выше характеристики во многом отражают эффективность процессора. Но есть еще целый ряд характеристик, описывающих внутреннюю архитектуру процессора, и сейчас мы рассмотрим некоторые важнейшие из них.

3.3.2 Шина данных процессора

Одной из самых общих характеристик процессора является разрядность его шины данных и шины адреса. (Вспомните, что такое шина).

Когда говорят о шине процессора, обычно имеют ввиду шину данных, которая является набором соединений, для передачи и приема данных. Чем больше сигналов одновременно

поступает на шину, тем больше данных по ней передается за определенный интервал времени, и тем быстрее она работает. Разрядность шины данных подобна количеству полос автомагистрали - чем больше полос, тем больше поток машин, чем шире шина данных, тем больше данных за одинаковые промежутки времени по ней передается. В процессоре 286 для приема и передачи двоичных данных используется 16 соединений, поэтому их шина данных считается 16-разрядной. У 32-х разрядных процессоров (например, 486), таких соединений вдвое больше, поэтому за единицу времени они передают и получают вдвое больше данных, чем 16-и разрядные процессоры - разумеется, эффективность выше. Современные процессоры (начиная с Pentium) имеют 64-х разрядную шину данных, поэтому они могут передавать в системную память по 64 бита за один такт.

Такая реализация позволяет ускорить обмен данными между быстрым процессором и относительно медленным ОЗУ при неизменной рабочей частоте последнего за счёт повышения пропускной способности шины данных (вспомните, что такое пропускная способность шины).

Таким образом мы определили, с помощью какой шины процессор связан с оперативной памятью и от разрядности этой шины конечно же зависит производительность процессора. Теперь давайте разберемся, как процессор обрабатывает полученные из оперативной памяти данные.

3.3.3. Шины и регистры процессора

Хоть процессор и получает данные из оперативной памяти с помощью шины некоторой ширины, это не значит, что внутри он может обрабатывать данные такой же разрядности. Давайте разберемся, как это происходит.

Количество битов данных, которые может обработать процессор за один прием, характеризуется разрядностью внутренних регистров.

Регистр - это по существу ячейка памяти внутри процессора, например, процессор может складывать числа, записанные в двух разных регистрах, а результат записывать в третий регистр. Разрядность регистров описывает разрядность обрабатываемых процессором данных. Разрядность регистров определяет также характеристики программного обеспечения и команд, выполняемых процессором. Например, процессоры с 32-разрядными внутренними регистрами могут выполнять 32-разрядные команды, которые обрабатывают данные 32-разрядными порциями, а процессоры с 16-разрядными регистрами этого делать не могут.

В современных процессорах для PC внутренние регистры являются 32 или 64 разрядными, причем 64-битных существует несколько вариантов – IA-64 для процессоров Itanium, . В некоторых старых процессорах разрядность внутренней шины данных (а шина состоит из линий передачи данных и регистров), больше, чем разрядность внешней шины данных, той, которую мы с Вами уже обсуждали (которая связывает процессор с оперативной памятью). Обычно такие процессоры являются более дешевыми вариантами своих старших собратьев. Такая архитектура (внутренняя шина и регистры вдвое шире внешней) позволяет проектировать и создавать, например, недорогие 16-разрядные материнские платы, устанавливая в них 32-разрядные процессоры, обеспечивая таким образом 32-разрядную совместимость процессора при 16 разрядном обмене с памятью. Но такой способ удешевления системы остался в прошлом и в настоящее время совершенно не применяется.

В современных процессорах все обстоит наоборот: внешняя шина данных, как мы уже говорили, 64-разрядная, а регистры и внутренняя шина процессора по-прежнему 32- разрядны. Странная ситуация, не правда ли? Но странной она кажется лишь до того момента, как мы узнаем, что в современном процессоре (например, Pentium) для обработки информации, поступающей по внешней 64-разрядной шине данных, ALU состоит из двух 32- разрядных блоков, которые называются конвейерами (подробнее в п.6), а 64-разрядная внешняя шина данных позволяет быстрее наполнить регистры процессора. Такая архитектура, применяющая для обработки поступивших данных несколько, называется суперскалярной и применяется сегодня во всех современных процессорах.

У процессоров с 64 разрядными регистрами ситуация с внешней и внутренней шинами несколько иная, например 64-битные процессоры фирмы AMD вообще имеют встроенный контроллер памяти, который работает с 64-битными модулями, и ширина шины памяти у них зависит от количества этих контроллеров. Шина адреса. Шина адреса представляет собой набор

проводников, по которым контроллеру памяти передается адрес ячейки памяти, в которую или из которой пересылаются данные. По каждому проводнику передается один бит адреса, соответствующий одной цифре в адресе. Увеличение количества проводников (разрядов шины) используемых для формирования адреса, позволяет увеличить количество адресуемых ячеек. Разрядность шины адреса определяет максимальный объем памяти, адресуемой процессором. В компьютерах применяется, как Вы знаете, двоичная системы счисления. Если, например, разрядность шины адреса составила бы всего один бит (один провод для передачи данных), то по этому проводу можно было бы передать всего два значения (логический ноль - нет напряжения, логическая единица - есть напряжение) и таким образом можно было бы адресоваться к двум ячейкам памяти. Такой бы процессор поддерживал обмен только с двумя байтами оперативной памяти! Использование двух бит для задания адреса позволило бы адресоваться уже к 4-м байтам памяти (00, 01, 10, 11 на шине - на четыре разных адреса можно указать). Вообще, количество разных значений, принимаемое n -разрядным двоичным числом равно 2 в степени n . Соответственно, при ширине шины адреса n бит, количество разных ячеек памяти, к которым можно адресоваться составляет 2 в степени n , поэтому говорят, что процессор поддерживает 2 в степени n байт оперативной памяти, или говорят, что адресное пространство процессора равно 2 в степени n байт. Например: процессор 8086 имел адресную шину 20 бит. Тогда он мог адресовать (2 в степени $20=1048576$) байт оперативной памяти, т.е. 1 Мбайт. Таким образом, максимальный объем оперативной памяти, поддерживаемой процессором 8086 составляет 1 Мбайт. 286-ой процессор имел адресную шину равную 24 битам, адресуя таким образом уже 16 Мбайт (обратите внимание: каждый новый бит в шине адреса, увеличивает объем адресуемой памяти вдвое, это естественно, если вспомнить формулу Объем памяти = 2 в степени разрядность шины). Современные процессоры имеют адресную шину равной 36 бит, что соответствует поддерживаемой оперативной памяти объемом 64 Гбайт!

Шины данных и адреса независимы, и разработчики микросхем выбирают их разрядность по своему усмотрению. Разрядность этих шин является показателем возможностей процессора: разрядность шины данных определяет возможности процессора быстро обмениваться информацией, разрядность адресной шины определяет объем поддерживаемой процессором памяти.

3.3.4. Частота и множитель процессора

Вы наверняка знаете, что современный процессор работает сегодня на частотах порядка 2...3 ГГц. С другой стороны оперативная память работает на сегодня на гораздо более низких частотах. Более того, не только оперативная память, но и чипсет работает на невысокой относительно процессора частоте. На материнской плате вообще нет компонентов, которые работали бы быстрее, чем системная шина и чипсет (точнее его северный мост). Впрочем, это и не удивительно.

Для того, чтобы, имея систему с 2,0 ГГц процессором, улучшить ее до 2,8 ГГц, нам достаточно лишь заменить процессор (естественно, учитывая при этом целый ряд факторов, но обо всем этом мы поговорим позже, давайте пока рассуждать упрощенно). Итак, вы извлекаете из процессорного гнезда 2,0 ГГц процессор и устанавливаете 2,8 ГГц. Вопрос: а материнская плата работает с новым процессором в прежнем режиме, в том же, в котором она работала и с процессором 2,0 ГГц? Если плата теперь работает втрое быстрее, то спрашивается: если для того, чтобы ускорить процессор его нужно заменить, значит, чтобы ускорить во столько же раз материнскую плату ее, наверное, тоже нужно заменить на втрое более производительную? Т.е. простой заменой процессора систему нельзя улучшить?

Нужно заменять материнскую плату, на новой плате должен быть другой, способный работать на утроенной частоте чипсет, нужно заменить и оперативную память? Если бы все было так, как мы сейчас описали, то компьютер вообще бы практически не подлежал бы улучшению (upgrade), а можно было бы лишь заменить его практически целиком.

Но это не так. В большинстве систем предел скорости на которой работает чипсет, составляет 800 МГц (200 МГц QDR). Тогда возникает главный вопрос: каким образом в ОДНОЙ И ТОЙ ЖЕ материнской плате могут работать и 2,0 ГГц и 2,8 ГГц процессора? Если частота

системной шины никогда не превышает 800 МГц, как процессор работает на более высокой частоте? Ответ кроется в том, что процессор НЕ работает на частоте системной шины (на этой частоте работает вся материнская плата, чипсет, память, но не процессор).

Процессор лишь использует для своей работы частоту системной шины. Дело в том, что процессор УМНОЖАЕТ эту частоту на некоторый множитель, таким образом получая результирующую частоту, на которой и работает. Например: Процессор, работающий на частоте 2,0 ГГц использует системную шину на частоте 200 МГц QDR, умножая ее на 10, а процессор, работающий на частоте 2,8 ГГц использует ту же частоту системной шины, умножая ее на 14.

Таким образом, в одну и ту же материнскую плату можно вставить различные процессоры, работающие на разных частотах за счет того, что частота системной шины используется одна и та же (или меньше, например: 2,0 ГГц процессора = 200 МГц системной шины \times 10), а процессор производит в себе умножение частоты поданной на него системной шины на некоторый, закрепленный в конкретном процессоре, множитель. Еще один важнейший момент.

Какая система быстрее? Процессор 2 ГГц = 100 МГц \times 20 или 2 ГГц = 200 МГц \times 10? Иными словами, влияет ли частота системной шины на производительность системы при неизменной частоте процессора? Естественно влияет. Разумеется, компьютер, у которого системная шина работает на частоте 200 МГц, при прочих равных условиях будет работать быстрее системы с тем же процессором, но на частоте системной шины 100 МГц.

Ведь и память, и прочие компоненты предпочтительнее использовать на более высокой частоте (естественно, если это позволяет их спецификация, если они, иными словами, для такой частоты предназначены). Т.е. компьютер производительнее не только потому, что в нем установлен более быстродействующий процессор; производительность так же зависит и от частоты системной шины, на которой, в свою очередь, работает оперативная память.

Важное замечание: естественно, производители чипсетов (а именно от чипсета зависит, какую частоту системной шины поддерживает материнская плата, от чипсета почти все зависит :)) постоянно стремятся увеличить частоту системной шины, поддерживаемую их продуктами, так как даже при очень быстром процессоре система не может быстрее обмениваться данными с памятью, чем позволяет системная шина и это ограничивает производительность всей системы. И, изучая с Вами в ближайшее время чипсеты, мы увидим, какое значение придается поддержке в сегодняшнем чипсете максимально возможной частоте системной шины и типу поддерживаемой памяти, так как, в конечном счете, борьба за быструю системную шины - это во многом борьба за ускорение обмена **память - процессор**.

3.3.5. Узлы динамического исполнения команд.

Как мы уже с вами говорили что бы компенсировать простой процессора в следствии снижения эффективности конвейера было создано три метода обработки информации в процессоре: **Предсказание ветвлений, анализ потока команд, упреждающее выполнение.**

Блок Предсказания ветвлений (branch prediction)

С его помощью можно выяснить, каким будет поток управления программы через несколько команд ветвления. При использовании специального механизма процессор может предсказать переходы или ветвления в потоке команд. В процессорах прежних поколений инструкция перехода приостанавливала конвейер (выборку инструкций) до исполнения собственно перехода, на чем, естественно, терялась производительность.

Предсказание переходов направляет поток выборки и декодирования по одной из ветвей. Статический метод предсказания работает по схеме, заложенной в процессор, считая, что переходы по одним условиям, вероятнее всего, произойдут, а по другим — нет. Динамическое предсказание опирается на предысторию вычислительного процесса — для каждого конкретного случая перехода накапливается статистика поведения, и переход предсказывается, основываясь именно на ней. Этот метод позволяет существенно ускорить выполнение программы, но при этом возникает проблема — в случае, если предсказание окажется неверным, всё содержимое конвейера тоже окажется неверным, и его придется очищать. А чем длиннее конвейер, тем больше будет потеря времени в результате его очистки, поэтому производители процессоров постоянно совершенствуют механизм предсказания ветвлений.

Анализ потока команд (out-of-order execution)

Это средство анализирует и планирует выполнение команд в оптимальной последовательности, независимо от их первоначального порядка в программе. Свойственный RISC-архитектуре, данный метод теперь реализуется и для процессоров x86. При этом изменяется порядок внутренних манипуляций данными, а внешние (шинные) операции ввода-вывода и записи в память выполняются, конечно же, в порядке, предписанном программным кодом. Однако эта способность процессора в наибольшей степени может блокироваться несовершенством программного кода (особенно 16-битных приложений), если он генерируется без учета возможности изменения порядка исполнения инструкций. Процессор рассматривает команды, из которых состоит выполняемая программа, и определяет, доступны ли они для обработки или же зависят от других команд, которые следует выполнить предварительно. Затем процессор определяет оптимальную последовательность обработки и выполняет команды наиболее эффективным способом.

Упреждающее выполнение (data forwarding)

Подразумевает начало исполнения инструкции до готовности всех операндов. При этом выполняются все возможные действия, и декодированная инструкция с одним операндом помещается в исполнительное устройство, где дожидается готовности второго операнда, выходящего с другого конвейера. С помощью этого метода процессор просматривает стоящие на очереди команды и выполняет те из них, к которым вероятно потребуются обратиться позже. Таким образом ряд команд процессор может выполнить заранее, а затем пользоваться результатами произведенных вычислений позже.

3.3.6. Предсказание ветвлений

В любой более-менее сложной программе присутствуют команды условного перехода: «Если некое условие истинно — перейти к исполнению одного участка кода, если нет — другого». С точки зрения скорости выполнения кода программы современным процессором, поддерживающим внеочередное исполнение, любая команда условного перехода — воистину бич божий. Ведь до тех пор, пока не станет известно, какой участок кода после условного перехода окажется «актуальным» — его невозможно начать декодировать и исполнять (см. внеочередное исполнение). Для того чтобы как-то примирить концепцию внеочередного исполнения с командами условного перехода, предназначается специальный блок: блок предсказания ветвлений. Как понятно из его названия, занимается он, по сути, «пророчествами»: пытается предсказать, на какой участок кода укажет команда условного перехода, ещё до того, как она будет исполнена. В соответствии с указаниями «штатного внутриядерного пророка», процессором производятся вполне реальные действия: «напророченный» участок кода загружается в кэш (если он там отсутствует), и даже начинается декодирование и выполнение его команд. Причём среди выполняемых команд также могут содержаться инструкции условного перехода, и их результаты тоже предсказываются, что порождает целую цепочку из пока не проверенных предсказаний! Разумеется, если блок предсказания ветвлений ошибся, вся проделанная в соответствии с его предсказаниями работа просто аннулируется.

На самом деле, алгоритмы, по которым работает блок предсказания ветвлений, вовсе не являются шедеврами искусственного интеллекта. Преимущественно они просты... и тупы. Ибо чаще всего команда условного перехода встречается в циклах: некий счётчик принимает значение X , и после каждого прохождения цикла значение счётчика уменьшается на единицу. Соответственно, до тех пор, пока значение счётчика больше нуля — осуществляется переход на начало цикла, а после того, как он становится равным нулю — исполнение продолжается дальше. Блок предсказания ветвлений просто анализирует результат выполнения команды условного перехода, и считает, что если N раз подряд результатом стал переход на определённый адрес — то и в $N+1$ случае будет осуществлён переход туда же. Однако, несмотря на весь примитивизм, данная схема работает просто замечательно: например, в случае, если счётчик принимает значение 100, а «порог срабатывания» предсказателя ветвлений (N) равен двум переходам подряд на один и тот же адрес — легко заметить, что 97 переходов из 98 будут предсказаны правильно!

Разумеется, несмотря на достаточно высокую эффективность простых алгоритмов, механизмы предсказания ветвлений в современных CPU всё равно постоянно совершенствуются и усложняются — но тут уже речь идёт о борьбе за единицы процентов: например, за то, чтобы повысить эффективность работы блока предсказания ветвлений с 95 процентов до 97, или даже с 97% до 99...

3.3.7 Блок предвыборки данных.

Блок предвыборки данных (**Prefetch**) очень похож по принципу своего действия на блок предсказания ветвлений — с той только разницей, что в данном случае речь идёт не о коде, а о данных. Общий принцип действия такой же: если встроенная схема анализа доступа к данным в ОЗУ решает, что к некоему участку памяти, ещё не загруженному в кэш, скоро будет осуществлён доступ — она даёт команду на загрузку данного участка памяти в кэш ещё до того, как он понадобится исполняемой программе. «Умно» (результативно) работающий блок предвыборки позволяет существенно сократить время доступа к нужным данным, и, соответственно, повысить скорость исполнения программы. К слову: грамотный Prefetch очень хорошо компенсирует высокую латентность подсистемы памяти, подгружая нужные данные в кэш, и тем самым, нивелируя задержки при доступе к ним, если бы они находились не в кэше, а в основном ОЗУ.

Однако, разумеется, в случае ошибки блока предвыборки данных, неизбежны негативные последствия: загружая де-факто «ненужные» данные в кэш, Prefetch вытесняет из него другие (быть может, как раз нужные). Кроме того, за счёт «предвосхищения» операции считывания, создаётся дополнительная нагрузка на контроллер памяти (де-факто, в случае ошибки — совершенно бесполезная).

Алгоритмы Prefetch, как и алгоритмы блока предсказания ветвлений, тоже не блещут интеллектуальностью: как правило, данный блок стремится отследить, не считывается ли информация из памяти с определённым «шагом» (по адресам), и на основании этого анализа пытается предсказать, с какого адреса будут считываться данные в процессе дальнейшей работы программы. Впрочем, как и в случае с блоком предсказания ветвлений, простота алгоритма вовсе не означает низкую эффективность: в среднем, блок предвыборки данных чаще «попадает», чем ошибается (и это, как и в предыдущем случае, прежде всего связано с тем, что «массированное» чтение данных из памяти, как правило происходит в процессе исполнения различных циклов).

3.3.8 Декодер

На самом деле, исполнительные блоки всех современных десктопных x86-процессоров... вовсе не работают с кодом в стандарте x86. У каждого процессора есть своя, «внутренняя» система команд, не имеющая ничего общего с теми командами (тем самым «кодом»), которые поступают извне. В общем случае, команды, исполняемые ядром — намного проще, «примитивнее», чем команды стандарта x86. Именно для того, чтобы процессор «внешне выглядел» как x86 CPU, и существует такой блок как декодер: он отвечает за преобразование «внешнего» x86-кода во «внутренние» команды, исполняемые ядром (при этом достаточно часто одна команда x86-кода преобразуется в несколько более простых «внутренних»). Декодер является очень важной частью современного процессора: от его быстродействия зависит то, насколько постоянным будет поток команд, поступающих на исполняющие блоки. Ведь они неспособны работать с кодом x86, поэтому то, будут они что-то делать, или простаивать — во многом зависит от скорости работы декодера. Достаточно необычный способ ускорить процесс декодирования команд реализовала в процессорах архитектуры NetBurst компания Intel — смотрите ниже про Trace cache. ☺

3.3.9 Исполняющие (функциональные) устройства

Пройдя через все уровни кэша и декодер, команды наконец-то попадают на те блоки, ради которых вся эта катавасия и устраивалась: исполняющие устройства. По сути, именно исполняющие устройства и являются единственным необходимым элементом процессора. Можно обойтись без кэша — скорость снизится, но программы работать будут. Можно обойтись без

декодера — исполняющие устройства станут сложнее, но работать процессор будет. В конце концов, ранние процессоры архитектуры x86 (i8086, i80186, 286, 386, 486, Am5x86) — как-то без декодера обходились. Без исполняющих устройств обойтись невозможно, ибо именно они исполняют код программы. В самом первом приближении они традиционно делятся на две больших группы: арифметико-логические устройства (ALU) и блок вычислений с плавающей точкой (FPU).

3.3.10 Арифметико-логические устройства (ALU)

ALU традиционно отвечают за два типа операций: арифметические действия (сложение, вычитание, умножение, деление) с целыми числами, логические операции с опять-таки целыми числами (логическое «и», логическое «или», «исключающее или», и тому подобные). Что, собственно, и следует из их названия. Блоков ALU в современных процессорах, как правило, несколько. Для чего — вы поймёте позже, прочитав раздел «Суперскалярность и внеочередное исполнение команд». Понятно, что ALU может исполнить только те команды, которые предназначены для него. Распределением команд, поступающих с декодера, по различным исполняющим устройствам, занимается специальный блок, но это уже, как говорится, «слишком сложные материи», и их вряд ли имеет смысл разъяснять в материале, который посвящен лишь поверхностному ознакомлению с основными принципами работы современных x86 CPU.

3.3.11 Блок вычислений с плавающей запятой

Согласно традиций русской математической школы, мы называем FPU «блоком вычислений с плавающей запятой», хотя буквально его название (Floating Point Unit) переводится как «...с плавающей точкой» — согласно американскому стандарту написания таких чисел.

FPU занимается выполнением команд, работающих с числами с плавающей запятой, кроме того, традиционно на него «вешают всех собак» в виде всяческих дополнительных наборов команд (MMX, 3DNow!, SSE, SSE2, SSE3...) — независимо от того, работают они с числами с плавающей запятой, или с целыми. Как и в случае с ALU, отдельных блоков в FPU может быть несколько, и они способны работать параллельно.

3.3.12 Сопроцессор, дополнительные наборы команд.

Набор команд x86, который мы с Вами уже обсуждали, имеет следующую особенность - он ориентирован на работу с целыми числами. Как быть, если процессору нужно извлечь квадратный корень, найти синус или логарифм? Естественно, что процессор может справиться с такой задачей, но, учитывая то, что он ориентирован на вычисления с целыми числами, выполнение такой операции займет у него много тактов.

В то же время Intel для всех своих процессоров разрабатывает так называемый сопроцессор - кристалл, который тоже умеет выполнять команды, но не x86, а другие, и поддерживаемый сопроцессором набор команд (называемый **x87**) ориентирован на работу с числами с плавающей запятой, таким образом, он (сoproцессор) перечисленные выше задачи как раз и призван решать. На заре развития PC считалось, что сопроцессор нужен небольшому количеству пользователей (действительно, зачем пользователю текстового редактора математические вычисления) и устанавливался в систему дополнительно; при желании пользователь мог отдельно приобрести сопроцессор и установить его в специальное гнездо на материнской плате. Сегодня ситуация полностью изменилась.

Сегодня сопроцессор нужен абсолютно всем пользователям: "в наш век мультимедиа" когда компьютер все больше и больше способен обрабатывать реалистичную 3-х мерную графику, математические расчеты становятся неотъемлемым атрибутом любого мультимедиа приложения (например, современной игры). Поэтому достаточно давно сопроцессор стали устанавливать вместе с процессором на один кристалл. Действительно, раз сопроцессор все равно нужен всем пользователям, то его разумно встроить в процессор, а не изготавливать отдельно: разумеется, это будет дешевле. На заре развития PC считалось, что сопроцессор нужен небольшому количеству пользователей (действительно, зачем пользователю текстового редактора математические

вычисления) и устанавливался в систему дополнительно; при желании пользователь мог отдельно приобрести сопроцессор и установить его в специальное гнездо на материнской плате. Сегодня ситуация полностью изменилась. Сегодня сопроцессор нужен абсолютно всем пользователям: "в наш век мультимедиа" когда компьютер все больше и больше способен обрабатывать реалистичную 3-х мерную графику, математические расчеты становятся неотъемлемым атрибутом любого мультимедиа приложения (например, современной игры). Поэтому достаточно давно сопроцессор стали устанавливать вместе с процессором на один кристалл. Действительно, раз сопроцессор все равно нужен всем пользователям, то его разумно встроить в процессор, а не изготавливать отдельно: разумеется, это будет дешевле.

Таким образом, любой современный процессор поддерживает два основных набора команд: **x86** и **x87**. Могут ли производители процессоров отказаться от этих наборов команд? Нет! Тогда получившаяся система уже не будет программно совместима с РС, так как программное обеспечение, написанное для РС на таком процессоре, уже не будет работать! Поддержка этих двух наборов команд - залог программной совместимости. А могут ли производители процессоров создавать и добавлять в процессор новые наборы команд? Конечно. Но что нужно для того, чтобы программы стали работать эффективнее на процессоре с новым набором команд? Будет ли у старого приложения, которое проектировали еще до разработки нового набора команд какие-то преимущества от исполнения на процессоре с новым набором команд? Нет! Ведь программа - это не что иное, как последовательность команд процессору. Если в программе нет ни одной новой команды (а откуда ей там взяться, если в момент написания программы новых команд еще не придумали), то естественно никакой пользы из того, что процессор может исполнять новые команды, старое приложение не извлечет. Поэтому, когда в процессор добавляют новый набор команд, нужно понимать, что пока разработчики программного обеспечения не начнут писать программы с учетом нового набора команд НИКАКОЙ пользы от него не будет. А старым программам от нового набора команд уже никогда не будет пользы (разве что авторы перепишут старую программу с учетом нового набора команд). Процессор, изначально, задумывался как универсальное устройство, которое могло с помощью изменения программы, решать любые задачи, будь то решение математических уравнений, игра, программа набора текста и др. Но со временем, к некоторым областям стали предъявляться особые требования по быстродействию. Для решения, некоторых узких задач, были введены дополнительные наборы команд - расширения основного набора x86.

Набор инструкций 3DNow!/Enhanced 3DNow!

Технология 3DNow! разработана фирмой AMD для процессоров K6-2, конкурента Pentium MMX и Pentium II. Направлена эта технология, как видно из названия, на ускорение работы с трехмерной графикой, а также другие вычисления связанные с потоковыми видео- и аудиоданными, хотя по сути является просто математическим набором команд. Основная цель, которую преследовала AMD при создании набора команд 3DNow! – создание альтернативы x87, т.к. сопроцессоры, которые применялись фирмой AMD для своих процессоров, были недостаточно производительными в сравнении с сопроцессорами Intel, разработчиком x87, а трёхмерная графика и игры являются основными потребителями математических команд. И если заметить, именно игры для большинства пользователей являются двигателем к выбору аппаратной части компьютера, поэтому AMD сделали рискованный маркетинговый ход – внедрили новый набор команд в свои процессоры, и начали активно преподносить свои процессоры как наиболее оптимизированные под игровой процесс. Как мы уже говорили, успех того или иного набора команд целиком зависит от его поддержки программным обеспечением, поэтому для успеха своих процессоров AMD сделали ставку именно на игры. Ее дальнейшее развитие – Enhanced 3DNow!, появилась в процессорах Athlon и Duron.

Технология MMX

В зависимости от контекста MMX может означать multi-media extensions (мультимедийные расширения) или matrix math extensions (матричные математические расширения). Технология MMX начала использоваться в старших моделях i487 процессоров Pentium пятого поколения в качестве расширения, благодаря которому ускоряется компрессия/декомпрессия видеоданных,

манипулирование изображением, шифрование и выполнение операций ввода-вывода - почти все операции, используемые во многих современных программах. MMX это расширение основного набора команд на 57 новых команд, а также во введении новой возможности выполнения команд, называемой одиночный поток команд - множественный поток данных (Single Instruction - Multiple Data, SIMD).

Набор инструкций SSE/SSE2/SSE3

SSE (Streaming SIMD Extensions - потоковые расширения SIMD), обновление технологии MMX, появились в Pentium III. Содержит 70 новых инструкций для работы с графикой и звуком в дополнение к существующим командам MMX. Инструкции SSE подобны инструкциям MMX и предварительно назывались MMX-2. Операции с плавающей точкой SSE реализованы в виде отдельного модуля в процессоре. Новые инструкции SSE позволяют более эффективно работать с трехмерной графикой, потоками аудио- и видеоданных, приложениями распознавания речи. Набор команд SSE2, который появились в Pentium 4 - это опять расширенный на новые 144 команды набор команд SSE, в первую очередь направлен на потоковые вычисления. И SSE3, появившийся в последних моделях Pentium 4, также представляет собой очередное расширение SSE.

3.3.13 Кэш – память

К важнейшим параметрам процессора относится так называемая кэш-память.

Представим себе, как происходит обмен информацией между процессором и памятью.

В большинстве случаев оперативная память не удовлетворяет потребностям современных процессоров в плане пропускной способности по данным, т.к. работает на значительно более низких частотах. Современный процессор работает на частотах до 3 ГГц. Естественно, что при обмене с памятью процессор достаточно долгое время будет ждать прихода новых порций данных и таким образом простаивать. Для того чтобы этого избежать, между памятью и процессором устанавливают дополнительно небольшой объем очень быстрой памяти, работающей без задержек на частоте процессора. Такая память и называется **кэш-память**, и если вы помните из предыдущего урока, она имеет тип **SRAM**. В современном процессоре встроено некоторое количество такой памяти (от 32 Кб до 3 Мб) и эта память обеспечивает снижение простоев процессора при операциях с оперативной памятью. Так кэш-память, которая установлена непосредственно в кристалле процессора, называется кэш-памятью первого уровня (или L1, Level1).

В переводе слово кэш (cache) означает «тайный склад», «тайник» («зачатка»). Тайна этого склада заключается в его «прозрачности» — для программы он не представляет собой дополнительной адресуемой области памяти. Кэш является дополнительным быстродействующим хранилищем копий блоков информации из основной памяти, вероятность обращения к которым в ближайшее время велика. Кэш не может хранить копию всей основной памяти, поскольку его объем во много раз меньше основной памяти.

Он хранит лишь ограниченное количество блоков данных и каталог (**cache directory**) — список их текущего соответствия областям основной памяти. Кроме того, кэшироваться может не вся память, доступная процессору.

При каждом обращении к памяти контроллер кэш-памяти по каталогу проверяет, есть ли действительная копия затребованных данных в кэше. Если она там есть, то При каждом обращении к памяти контроллер кэш-памяти по каталогу проверяет, есть ли действительная копия затребованных данных в кэше. Если она там есть, то это случай **кэш-попадания (cache hit)**, и данные берутся из кэш-памяти. Если действительной копии там нет, это случай **кэш-промаха (cache miss)**, и данные берутся из основной памяти. В соответствии с алгоритмом кэширования блок данных, считанный из основной памяти, при определенных условиях заместит один из блоков кэша. От интеллектуальности алгоритма замещения зависит процент попаданий и, следовательно, эффективность кэширования.

Поиск блока в списке должен производиться достаточно быстро, чтобы «задумчивостью» в принятии решения не свести на нет выигрыш от применения быстродействующей памяти. Обращение к основной памяти может начинаться одновременно с поиском в каталоге, а в случае

попадания — прерываться (архитектура **Look aside**). Это экономит время, но лишние обращения к основной памяти ведут к увеличению энергопотребления. Другой вариант: обращение к внешней памяти начинается только после фиксации промаха (архитектура **Look Through**), при этом теряется по крайней мере один такт процессора, зато экономится энергия.

В современных компьютерах кэш обычно строится по двухуровневой схеме.

Первичный кэш (L1 Cache) Или Кеш первого уровня, встроен во все процессоры начиная с 486, это внутренний кэш. Объем этого кэша невелик (от 8 до 64 Кбайт). Для повышения производительности для данных и команд часто используется отдельный кэш (так называемая Гарвардская архитектура — противоположность Принстонской, использующей общую память для команд и данных).

Вторичный кэш (L2 Cache) Или Кеш второго уровня, для процессоров 486 и Pentium является внешним, т.е. устанавливается на системной плате, его чипы выделены на рисунке слева, а в Pentium pro и всех последующих процессорах располагается в одной упаковке с ядром и подключается специальной внутренней шиной к процессору, или вообще является частью кристалла процессора, на рисунке справа – фотография кристалла процессора, и выделен кэш 2-го уровня.

Разница в расположении кэша очень существенная – в случае, если он находится на материнской плате, он работает на частоте системной (внешней) шины процессора, и доступ к нему осуществляется опять же через эту шину. А в случае с подключением КЭШа отдельной шиной, частота его работы и производительность этой шины не будет зависеть от системной шины, соответственно и производительность будет намного выше.

Кэш-контроллер должен обеспечивать **когерентность (coherency)** — согласованность данных кэш-памяти обоих уровней с данными в основной памяти, при том условии, что обращение к этим данным может производиться не только процессором, но и другими активными (busmaster) адаптерами, подключенными к шинам (PCI, VLB, ISA и т. д.).

Следует также учесть, что процессоров может быть несколько, и у каждого может быть свой внутренний кэш. Контроллер кэша оперирует **строками (cache line)** фиксированной длины. Строка может хранить копию блока основной памяти, размер которого, естественно, совпадает с длиной строки. С каждой строкой кэша связана информация об адресе скопированного в нее блока основной памяти и об ее состоянии. Строка может быть действительной (valid) — это означает, что в текущий момент времени она достоверно отражает соответствующий блок основной памяти, или недействительной. Информация о том какой именно блок занимает данную строку (то есть старшая часть адреса или номер страницы), и о ее состоянии называется **тегом (tag)** и хранится в связанной с данной строкой ячейке специальной **памяти тегов (tag RAM)**. В операциях обмена с основной памятью обычно строка участвует целиком (несекторизованный кэш), для процессоров 486 и старше длина строки совпадает с объемом данных, передаваемых за один пакетный цикл (для 486 это $4 \times 4 = 16$ байт, для Pentium $4 \times 8 = 32$ байт). Возможен и вариант секторизованного (sectored) кэша, при котором одна строка содержит несколько смежных ячеек — секторов, размер которых соответствует минимальной порции обмена данных кэша с основной памятью. При этом в записи каталога, соответствующей каждой строке, должны рваться биты действительности для каждого сектора данной строки. Секторизованное позволяет экономить память, необходимую для рвления каталога при увеличении объема кэша, поскольку большее количество бит каталога отводится под тег и выгоднее использовать дополнительные биты действительности, чем увеличивать глубину индекса (количество элементов) каталога.

Строки кэша под отображение блока памяти выделяются при промахах операций чтения, в Р6 строки заполняются и при записи. Запись блока, не имеющего копии в кэше, производится в основную память (для повышения быстродействия запись может производиться через буфер отложенной записи). Поведение кэш-контроллера при операции записи в память, когда копия затребованной области находится в некоторой строке кэша, определяется его алгоритмом, или политикой записи (**Write Policy**). Существуют две основные политики записи данных из кэша в основную память: сквозная запись **WT (Write Through)** и обратная запись **WB (Write Back)**.

Политика WT предусматривает выполнение каждой операции записи (даже однобайтной), попадающей в кэшированный блок, одновременно и в строку кэша, и в основную память. При этом процессору при каждой операции записи придется выполнять относительно длительную запись в основную память. Алгоритм достаточно прост в реализации и легко обеспечивает

целостность данных за счет постоянного совпадения копий данных в кэше и основной памяти. Для него не нужно хранить признаки присутствия и модифицированности — вполне достаточно только информации тега (при этом считается, что любая строка всегда отражает какой-либо блок, а какой именно — указывает тег). Но эта простота оборачивается низкой эффективностью записи. Существуют варианты этого алгоритма с применением отложенной буферизованной записи, при которой данные в основную память переписываются через FIFO-буфер во время свободных тактов шины.

Политика WB позволяет уменьшить количество операций записи на шине основной памяти. Если блок памяти, в который должна производиться запись, отображен в кэше, то физическая запись сначала будет произведена в эту действительную строку кэша, и она будет отмечена как грязная (dirty), или модифицированная, то есть требующая выгрузки в основную память. Только после этой выгрузки (записи в основную память) строка станет чистой (clean), и ее можно будет использовать для кэширования других блоков без потери целостности данных. В основную память данные переписываются только целой строкой. Эта выгрузка контроллером может откладываться до наступления крайней необходимости (обращение к кэшированной памяти другим абонентом, замещение в кэше новыми данными) или выполняться в свободное время после модификации всей строки.

Эксклюзивный и не эксклюзивный кэш

Концепции эксклюзивного и не эксклюзивного кэширования очень просты: в случае не эксклюзивного кэша, информация на всех уровнях кэширования может дублироваться. Таким образом, L2 может содержать в себе данные, которые уже находятся в L1I и L1D, а L3 (если он есть) может содержать в себе полную копию всего содержимого L2 (и, соответственно, L1I и L1D). Эксклюзивный кэш, в отличие от не эксклюзивного, предусматривает чёткое разграничение: если информация содержится на каком-то уровне кэша — то на всех остальных она отсутствует. Плюс эксклюзивного кэша очевиден: общий размер кэшируемой информации в данном случае равен суммарному объёму кэшей всех уровней — в отличие от не эксклюзивного кэша, где размер кэшируемой информации (в худшем случае) равен объёму самого большого уровня кэша. Минус эксклюзивного кэша менее очевиден, но он есть: необходим специальный механизм, который следит за собственно «эксклюзивностью» (так, например, при удалении информации из L1-кэша, перед этим автоматически инициируется процесс её копирования в L2).

Не эксклюзивный кэш традиционно использует компания Intel, эксклюзивный (с момента появления процессоров Athlon на ядре Thunderbird) — компания AMD. В целом, мы наблюдаем здесь классическое противостояние между объёмом и скоростью: за счёт эксклюзивности, при одинаковых объёмах L1/L2 у AMD общий размер кэшируемой информации получается больше — но за счёт неё же он работает медленней (задержки, вызванные наличием механизма обеспечения эксклюзивности). Следует, наверное, заметить, что недостатки не эксклюзивного кэша компания Intel в последнее время компенсирует просто, тупо, но весомо: наращивая его объёмы. Для топовых процессоров данной компании стал уже почти что нормой L2-кэш объёмом 2 МБ — и AMD с её 128 КБ L1C+L1D и максимум 1 МБ L2 пока «не переплюнуть» эти 2 МБ даже за счёт эксклюзивности.

Кроме того, увеличивать общий объём кэшируемой информации за счёт введения эксклюзивной архитектуры кэша имеет смысл только в том случае, когда выигрыш в объёме получается достаточно большим. Для компании AMD это актуально т.к. у её сегодняшних CPU суммарный объём L1D+L1I равен 128 КБ. Процессорам Intel, у которых объём L1D равен максимум 32 КБ, а L1I иногда имеет совсем другую структуру (см. про Trace cache), введение эксклюзивной архитектуры дало бы намного меньше пользы.

***Замечание:** есть такое распространённое заблуждение, что архитектура кэша у CPU компании Intel «инклюзивная». На самом деле — нет. Именно НЕ эксклюзивная. Инклюзивная архитектура предусматривает, что на «нижнем» уровне кэша не может находиться ничего, чего нет на более «верхнем». Не эксклюзивная архитектура всего лишь допускает дублирование данных на разных уровнях.*

Trace cache

Концепция Trace cache, состоит в том, чтобы сохранять в кэше инструкций первого уровня (L1) не те команды, которые считаны из памяти, а уже декодированные последовательности (см. декодер). Таким образом, если некая x86-команда выполняется повторно, и она всё ещё находится в L1, декодеру процессора не нужно снова преобразовывать её в последовательность команд «внутреннего кода», так как L1 содержит данную последовательность в уже декодированном виде. Концепция Trace cache очень удачно вписывается в общую концепцию архитектуры Intel NetBurst, ориентированную на создание процессоров с очень высокой частотой работы ядра. Однако полезность Trace cache для [относительно] менее высокочастотных CPU до сих пор находится под вопросом, так как сложность организации Trace cache становится сопоставима с задачей конструирования обычного быстрого декодера. Поэтому, отдавая должное оригинальности идеи, мы всё же сказали бы, что универсальным решением «на все случаи жизни» Trace cache считать нельзя.

3.3.14 Процессорное ядро

Различия между ядрами одной микроархитектуры

«Процессорное ядро» (как правило, для краткости его называют просто «ядро») — это конкретное воплощение [микро]архитектуры (т.е. «архитектуры в аппаратном смысле этого слова»), являющееся стандартом для целой серии процессоров. Например, NetBurst — это микроархитектура, которая лежит в основе многих сегодняшних процессоров Intel: Celeron, Pentium 4, Xeon. Микроархитектура задает общие принципы: длинный конвейер, использование определенной разновидности кэша кода первого уровня (Trace cache), прочие «глобальные» особенности. Ядро — более конкретное воплощение. Например, процессоры микроархитектуры NetBurst с шиной 400 МГц, кэшем второго уровня 256 килобайт, и без поддержки Hyper-Threading — это более-менее полное описание ядра Willamette. А вот ядро Northwood имеет кэш второго уровня уже 512 килобайт, хотя также основано на NetBurst. Ядро AMD Thunderbird основано на микроархитектуре K7, но не поддерживает набор команд SSE, а вот ядро Palomino — уже поддерживает.

Таким образом, можно сказать что «ядро» — это конкретное воплощение определенной микроархитектуры «в кремнии», обладающее (в отличие от самой микроархитектуры) определенным набором строго обусловленных характеристик. Микроархитектура — аморфна, она описывает общие принципы построения процессора. Ядро — конкретно, это микроархитектура, «обросшая» всевозможными параметрами и характеристиками. Чрезвычайно редки случаи, когда процессоры сменяли микроархитектуру, сохраняя название. И, наоборот, практически любое наименование процессора хотя бы несколько раз за время своего существования «меняло» ядро. Например, общее название серии процессоров AMD — «Athlon XP» — это одна микроархитектура (K7), но целых четыре ядра (Palomino, Thoroughbred, Barton, Thorton). Разные ядра, построенные на одной микроархитектуре, могут иметь, в том числе разное быстродействие.

Ревизии

Ревизия — одна из модификаций ядра, крайне незначительно отличающаяся от предыдущей, почему и не заслуживает звания «нового ядра». Как правило, из выпусков очередной ревизии производители процессоров не делают большого события, это происходит «в рабочем порядке». Так что даже если вы покупаете один и тот же процессор, с полностью аналогичным названием и характеристиками, но с интервалом где-то в полгода — вполне возможно, фактически он будет уже немного другой. Выпуск новой ревизии, как правило, связан с какими-то мелкими усовершенствованиями. Например, удалось чуть-чуть снизить энергопотребление, или понизить напряжение питания, или еще что-то оптимизировать, или была устранена пара мелких ошибок. С точки зрения производительности мы не помним ни одного примера, когда бы одна ревизия ядра отличалась от другой настолько существенно, чтобы об этом имело смысл говорить. Хотя чисто теоретически возможен и такой вариант — например, подвергся оптимизации один из блоков процессора, ответственный за исполнение нескольких команд. Подводя итог, можно сказать что «заморачиваться» ревизиями процессоров чаще всего не стоит: в очень редких случаях изменение

ревизии вносит какие-то кардинальные изменения в процессор. Достаточно просто знать, что есть такая штука — исключительно для общего развития.

Частота работы ядра

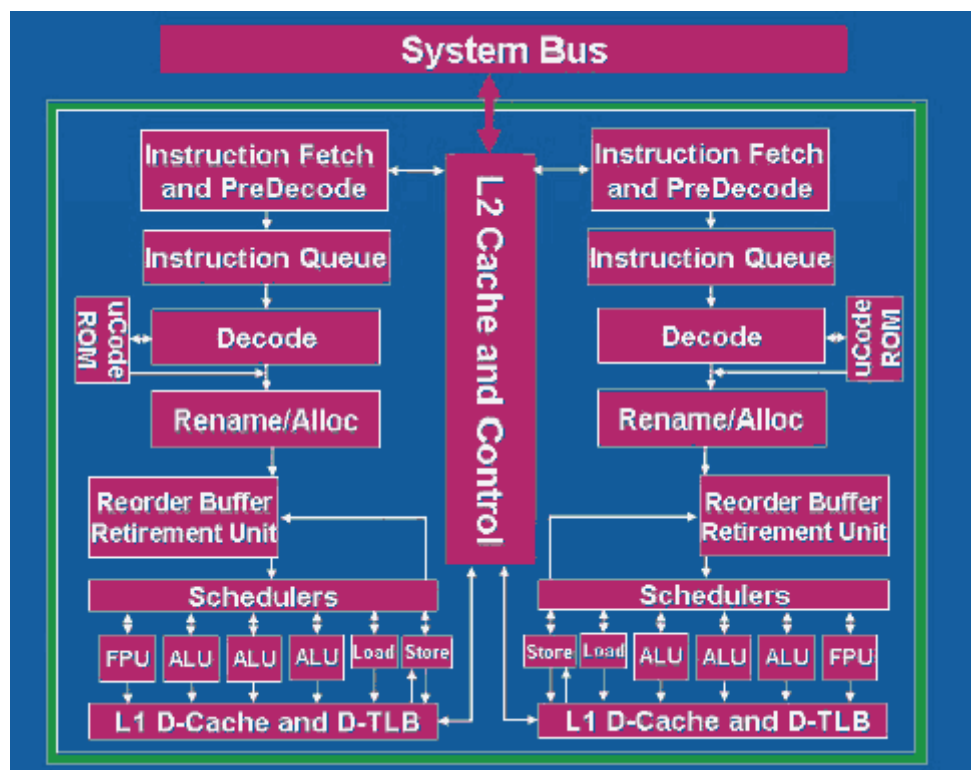
Как правило, именно этот параметр в просторечии именуют «частотой процессора». Хотя в общем случае определение «частота работы ядра» всё же более корректно, так как совершенно не обязательно все составляющие CPU функционируют на той же частоте, что и ядро (наиболее частым примером обратного являлись старые «слотовые» x86 CPU — Intel Pentium II и Pentium III для Slot 1, AMD Athlon для Slot A — у них L2-кэш функционировал на 1/2, и даже иногда на 1/3 частоты работы ядра). Ещё одним распространённым заблуждением является уверенность в том, что частота работы ядра однозначным образом определяет производительность. На самом деле это дважды не так: во-первых, каждое конкретное процессорное ядро (в зависимости от того, как оно спроектировано, сколько содержит исполняющих блоков различных типов, и т.д. и т.п.) может исполнять различное количество команд за один такт, частота же — это всего лишь количество таких тактов в секунду. Таким образом (приведенное далее сравнение, разумеется, очень сильно упрощено и поэтому весьма условно) процессор, ядро которого исполняет 3 инструкции за такт, может иметь на треть меньшую частоту, чем процессор, исполняющий 2 инструкции за такт — и при этом обладать полностью аналогичным быстродействием.

Во-вторых, даже в рамках одного и того же ядра, увеличение частоты вовсе не всегда приводит к пропорциональному увеличению быстродействия. Здесь вам очень пригодятся знания, которые вы могли почерпнуть из раздела «1. Общие принципы обработки данных».

Дело в том, что скорость исполнения команд ядром процессора — это вовсе не единственный показатель, влияющий на скорость выполнения программы. Не менее важна скорость поступления команд и данных на CPU. Представим себе чисто теоретически такую систему: быстродействие процессора — 10'000 команд в секунду, скорость работы памяти — 1000 байт в секунду. Вопрос: даже если принять, что одна команда занимает не более одного байта, а данных у нас нет совсем, с какой скоростью будет исполняться программа в такой системе? Правильно: не более 1000 команд в секунду, и производительность CPU тут совершенно ни при чём: мы будем ограничены не ей, а скоростью поступления команд в процессор. Таким образом, следует понимать: невозможно непрерывно наращивать одну только частоту ядра, не ускоряя одновременно подсистему памяти, так как в этом случае начиная с определённого этапа, увеличение частоты CPU перестанет сказываться на увеличении быстродействия системы в целом.

3.3.15 Блок схема узлов процессора

Теперь давайте попробуем объединить все выше перечисленные модули блоки и узлы в одну единую схему, и посмотрим что у нас с вами может получиться, если за основу взять двоядерный процессор, а мы с вами получим приблизительно следующее



3.4. Режимы работы процессора.

Все 32-разрядные и более поздние процессоры Intel, начиная с 386, могут выполнять программы в нескольких режимах. Режимы процессора предназначены для выполнения программ в различных средах; в разных режимах работы возможности чипа не одинаковы, потому, что команды выполняются по-разному. В зависимости от режима процессора изменяется схема управления памятью системы и задачами.

Процессоры могут работать в трех режимах: реальном, защищенном и виртуальном реальном режиме (реальном внутри защищенного).

3.4.1 Реальный режим.

В первоначальном IBM PC использовался процессор 8088, который мог выполнять 16-разрядные команды, используя 16-разрядные внутренние регистры и адресовать только 1 Мб памяти, используя 20-и разрядную шину адреса. Все программное обеспечение PC первоначально было предназначено для этого процессора, оно было разработано на основе 16-разрядной системы команд и модели памяти, объемом 1 Мб. Например DOS, все программное обеспечение DOS написано в расчете на 16-разрядные команды. Более поздние процессоры, например 286, могли также выполнять те же самые 16- разрядные команды, что и первоначальный 8088, но намного быстрее.

Другими словами процессор 286 был полностью совместим с первоначальным 8088. 16-разрядный режим, в котором выполнялись команды процессоров 8088 и 80286 был назван реальным режимом. Все программы, выполняющиеся в реальном режиме, должны использовать только 16- разрядные команды и 20-разрядный адрес. Для программного обеспечения такого типа используется однозадачный режим, т.е. одновременно должна выполняться только одна программа. Нет никакой встроенной защиты для предотвращения перезаписи ячеек памяти, занятых одной программой или даже самой операционной системой, другими программами: Это означает, что при выполнении нескольких программ вполне могут быть испорчены данные или код одной из программ, что может привести к остановке системы.

3.4.2 Защищенный режим

Первым 32-разрядным процессором, предназначенным для PC, был 386-ой. Этот чип мог выполнять абсолютно новую 32-разрядную систему команд. Для того чтобы полностью использовать преимущество этой новой системы команд, были необходимы 32-разрядная операционная система и 32-разрядные приложения. Этот новый режим называли защищенным, так как выполняющиеся в нем программы защищены от перезаписи используемых ими областей памяти другими программами.

Такая защита делает систему более надежной, так как уже ни одна программа с ошибками не сможет повредить другие программы или операционную систему. Зная, что разработка новых операционных систем и приложений, использующих преимущество 32-разрядного защищенного режима, займет некоторое время, Intel предусмотрела в процессоре 386 обратно совместимый реальный режим. Благодаря этому процессор 386 мог выполнять обычные 16-разрядные приложения и операционные системы. Причем они выполнялись намного быстрее, чем на любом процессоре предыдущего поколения. Для большинства пользователей этого было достаточно: они не востребовали 32-разрядные системы и приложения и довольствовались тем, что уже имеющиеся у них 16-разрядные программы работали быстрее. К сожалению, из-за этого 386 процессор так никогда и не использовался в защищенном режиме, и, стало быть, все преимущества такого режима терялись. Когда современный высокопроизводительный процессор работает в реальном режиме, то он напоминает чудовищно ускоренный 8088! Т.е процессор может хоть и с огромной (по сравнению с оригинальным 8088) скоростью, но все же выполнять только 16-разрядные приложения и адресоваться только к памяти размером 1 Мб, с которой мог работать 8088. Поэтому были необходимы новые операционные системы и новые приложения, которые могли бы на современных процессорах выполняться в защищенном 32-разрядном режиме. Однако пользователи сопротивлялись всем попыткам перехода к 32-разрядной среде. Для них это означало, что нужно, по крайней мере частично, отказываться от старого программного обеспечения

Только в августе 1995 года (спустя 10 лет после выхода первого 32-разрядного процессора) наконец появилась первая пользовательская 32-разрядная операционная система Windows 95, да и то, пользователи приняли ее во многом потому, что она частично 16-разрядная и поэтому без труда исполняет как новые 32-разрядные программы, так и старые, 16-разрядные. Именно для такой обратной совместимости Windows 95 использовала третий режим процессора:

3.4.3 Виртуальный реальный режим

Виртуальный реальный, по существу, является режимом выполнения 16-разрядной среды (реальный режим), который реализован внутри 32-разрядного защищенного режима. Выполняя команды в окне DOS в Windows 95/98, вы создаете виртуальный сеанс реального режима. Поскольку защищенный режим является подлинно многозадачным, фактически можно выполнять несколько сеансов реального режима, причем в каждом сеансе собственное программное обеспечение выполняется на собственном виртуальном компьютере. И все эти приложения могут выполняться одновременно, даже во время выполнения других 32-разрядных программ.

Следует обратить внимание на то, что любая программа, выполняющаяся в виртуальном реальном режиме, может обращаться к памяти, объемом до 1 Мб, причем для каждой такой программы это будет как бы первый и единственный мегабайт памяти в системе. Виртуальное реальное окно полностью имитирует среду процессора 8088и если не учитывать быстродействие, программное обеспечение в виртуальном реальном режиме выполняется так, как выполнялось бы на самых первых PC в реальном режиме. При запуске каждого 16-разрядного приложения Windows 95/98 создает так называемую виртуальную машину DOS, выдает ей 1 Мб памяти и на этой машине 16-разрядное приложение выполняется. Следует обратить внимание на то, что все процессоры при включении начинают работать в реальном режиме, и только при старте 32-разрядной операционной системы происходит переключение в 32-разрядный режим.

Так же следует отметить, что не любое 16-разрядное приложение будет корректно работать в виртуальном реальном режиме. Например, диагностические программы для обслуживания аппаратного обеспечения делают вещи, не предусмотренные в виртуальном реальном режиме (в

первую очередь пытаются напрямую работать с аппаратурой). Такие программы нельзя запускать из Windows 95/98, для запуска таких программ необходимо стартовать компьютер с операционной системой DOS и выполнять эти приложения в настоящем, а не виртуальном реальном режиме.

3.4.4 64 разрядный расширенный режим IA 32e (AMD64, x86-64, EM64T)

Этот режим является расширением архитектуры IA-32, разработанным компанией AMD и в дальнейшем поддержанным Intel. Процессоры, поддерживающие 64-разрядные расширения, могут работать в реальном режиме (8086), режиме IA-32 или IA-32e. При использовании режима IA-32 процессор может работать в защищенном или виртуальном реальном режиме.

Режим IA-32e позволяет работать в 64_разрядном режиме или в режиме совместимости, что подразумевает возможность одновременного выполнения 64 и 32разрядных приложений.

Режим IA-32e включает в себя два подрежима.

- **64 разрядный режим.** Позволяет 64_разрядной операционной системе выполнять 64 - разрядные приложения.
- **Режим совместимости.** Позволяет 64_разрядной операционной системе выполнять 32 - разрядные приложения.

Первый подрежим активизируется после загрузки 64-разрядной операционной системы и используется 64-разрядными приложениями. В 64-разрядном под режиме доступно несколько новых функций.

- 64_разрядная линейная адресация памяти.
- Поддержка физической памяти объемом более 4 Гбайт (определенные ограничения накладываются процессором).
- Восемь новых регистров общего назначения GPR (general_purpose register).
- Восемь новых регистров для поточных расширений SIMD (MMX, SSE, SSE2 и SSE3).
- 64 - разрядные регистры GPR и указатели инструкций.

Режим совместимости IE - 32e позволяет запускать 32 и 16_разрядные приложения под управлением 64 - разрядной операционной системы. К сожалению, наследуемые 16 - разрядные программы, работающие в виртуальном реальном режиме (например, приложения DOS), не поддерживаются, а значит, их выполнение невозможно. Данное ограничение наверняка будет представлять наибольшую проблему для пользователей. Подобно 64 - разрядному режиму, режим совместимости активизируется операционной системой для отдельных приложений, благодаря чему становится возможным одновременное выполнение 64_ и 32_разрядных приложений.

Для того чтобы все эти приложения работали, необходима 64_разрядная операционная система и, что гораздо важнее, 64 - разрядные драйверы для всех устройств, предназначенные именно для этой операционной системы.

Рассматривая возможность перехода от 32 к 64 - разрядной технологии, следует принимать во внимание поддерживаемый объем памяти, доступность драйверов и совместимость программного обеспечения. Как уже отмечалось, переход от 16 - разрядных вычислений к 32 - разрядным занял 16 лет. Конечно, переход от 32 - разрядных вычислений к 64 - разрядным займет не 16 лет, но 2–3 года на это все же потребуется.

3.5. Многопроцессорные системы

Довольно давно возникла идея использования нескольких процессоров, вместо одного. Т.е. руководствуясь принципом «одна голова хорошо, а две лучше», производители процессоров решили найти способ увеличения производительности компьютера, не изменяя при этом параметров самих процессоров. Таким образом, стали появляться системы с 2-мя, 4-мя и более

процессорами. По идее производительность таких систем должна быть больше в 2, 4 и т.д. раз, но на практике это совсем не так.

При создании многопроцессорных систем надо учитывать множество условий:

- Материнская плата должна уметь работать с несколькими процессорами, т.е. на ней должен быть соответствующий чипсет, и необходимое число разъёмов для процессоров.
- Сам процессор должен поддерживать работу в многопроцессорных системах, такие современные процессоры часто имеют в маркировке индекс **MP**.

Операционные системы и программное обеспечение также должны уметь работать с несколькими процессорами, в следующем курсе, мы будем ещё упоминать об этом, когда будем рассматривать возможности операционных систем. Многопроцессорные системы дают хороший прирост производительности, только тогда, когда на них выполняется специализированное программное обеспечение.

Компьютер с несколькими процессорами работает под управлением операционной системы, которая может распределять разные задачи по разным процессорам компьютера. А программы, которые выполняются на таком компьютере, должны состоять из нескольких потоков, которые можно выполнять независимо друг от друга. За счет этих возможностей будет увеличиваться производительность. Соответственно, если операционная система и программное обеспечение не отвечает этим требованиям, то и выигрыша от применения многопроцессорной системы не будет никакого, и её производительность будет такой же, что и у однопроцессорной системы. Существуют два режима работы многопроцессорных систем – симметричный и несимметричный. Рассмотрим их подробнее:

- **AMP, Asymmetric Multiprocessing.** Асимметричная многопроцессорность. В этом режиме операционная система выполняется на одних процессорах, а приложения выполняются другими процессорами. Такой режим в определённых условиях является неэффективным, т.к. нагрузка распределяется неравномерно, поэтому сейчас чаще всего применяется второй режим.

- **SMP, Symmetric Multiprocessing.** Симметричная многопроцессорность. В этом режиме и операционная система, и приложения выполняются любым процессором, в зависимости от его загрузки, что обеспечивает большую гибкость при распределении нагрузки, и соответственно большую производительность. Для работы в таком режиме был разработан специальный протокол, который называется **APIC**.

У многопроцессорных систем есть кроме множества достоинств, и явный недостаток – стоимость. Материнские платы для многопроцессорных систем стоят намного дороже однопроцессорных, да и покупка нескольких процессоров тоже не является дешевым удовольствием. Поэтому производители стали разрабатывать более дешевые варианты многопроцессорности.

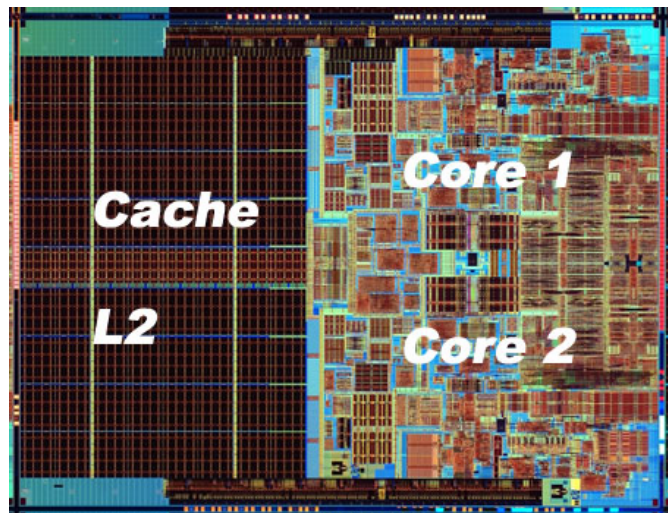
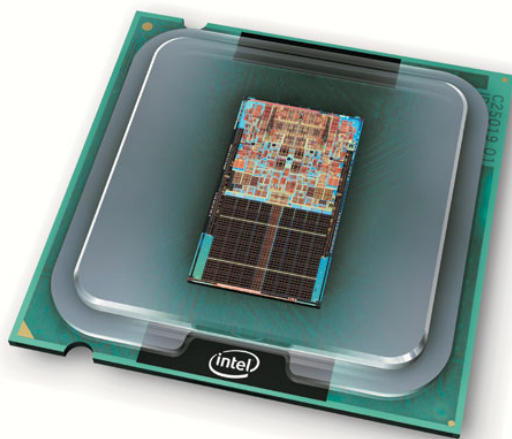
Первый из этих вариантов, который появился в процессорах Pentium 4 на ядре Northwood, это технология **Hyper Threading (HT)**, которая представляет из себя "логическую многопроцессорность". В чём её особенность?

Как вы уже знаете, процессор состоит из нескольких блоков, выполняющих различные задачи, например 2 конвейера ALU, из которых один основной а второй вспомогательный, т.е. загружен не всегда, 1 FPU и блок загружающий и выгружающий команды и данные из памяти. Представьте ситуацию, что во время выполнения программы возникает ситуация, когда на протяжении работы нескольких стадий конвейера загружены 60% только ALU, FPU не загружен вообще и блок загрузки-выгрузки занят на 30%. В среднем выйдет процент использования ресурсов процессора примерно 30%. Но это только конкретный случай, возможны и масса других ситуаций, когда процессор будет использовать большее, или меньшее число своих блоков, и во многом это зависит от программного обеспечения. Сама Intel заявляет, что блоки процессора Pentium 4 загружены в среднем не более чем на 35%. Поэтому они и предложили технологию, которая при небольших изменениях ядра (увеличение примерно на 10%), позволяет использовать простаивающие блоки процессора в качестве дополнительного, 2-го процессора. Для этого был добавлен ещё один блок регистров, и программы работают с таким процессором не как с одним, и как с двумя. Конечно, такой подход не даст полноценной альтернативы многопроцессорным системам, но в зависимости от конкретного применения HT, прирост

производительности может достигать 30%, хотя для домашнего пользователя он может быть равным и 0%, т.к. задачи выполняемые ими, как правило не вообще требуют многопроцессорности.

Также, с уменьшением технологического процесса производства процессоров, появилась возможность создавать на одном кристалле, который помещается в стандартный корпус процессора 2 и более процессоров. Такие процессоры называются **многоядерными**. А такую архитектуру называют **CMP, Cellular MultiProcessing**, сотовая мультипроцессорность. Особенностью, и отличием такой архитектуры от SMP, является использование несколькими ядрами общего пространства памяти.

Производители процессоров в настоящее время выпускают 2-ядерные процессоры, и в их планах на ближайшее будущее выпуск процессоров с 4-мя ядрами и более. На фото изображение двухъядерного процессора Pentium, и можно чётко различить два одинаковых ядра.



3.6 История развития процессоров (дополнительный материал, не обязательно)

Теперь, разобравшись со всеми терминами и понятиями, с которыми нам придется столкнуться, давайте начнем изучать собственно сами реальные процессоры. Стоит ли подробно изучать всю процессорную продукцию, появившуюся на рынке РС начиная с 1977 года? Наверное, нет. На мой взгляд, подробно следует изучить лишь те системы, которые сейчас присутствуют на рынке. И, естественно, мы уделим самое пристальное внимание таким процессорам. Но, тем не менее, мне кажется, что все же необходимо, хотя бы для общего развития и представления тенденций на рынке, сделать небольшой обзор всей истории развития процессоров, применяемых в РС.

Итак, давайте начнем.

3.6.1. Первое поколение процессоров.

Процессоры i8086 и i8088



В июне 1977 года Intel представила свой новый процессор **8086**. Процессор 8086 был полностью 16-разрядным, т.е. внутренние регистры, внутренняя и внешняя шины данных были 16-разрядными. Разрядность шины адреса составляла 20 бит, и он мог адресовать память

емкостью 1 Мбайт. Но как оказалось, многие пользователи не могут позволить себе иметь полностью 16-разрядный процессор т.к. его цена была чрезмерно высока, и через некоторое время была представлена усеченная версия процессора 8086, названная **8088**. В ней, по существу, были удалены 8 из 16 разрядов на шине данных, и теперь процессор 8088 мог рассматриваться как 8-разрядная микросхема с точки зрения ввода и вывода данных. Однако, поскольку в нем были полностью сохранены 16-разрядные внутренние регистры и 20-разрядная шина адреса, процессор 8088 так же выполнял 16-разрядное программное обеспечение и адресовал 1 Мбайт оперативной памяти.

В 1981 году IBM начинает выпуск компьютеров, которые называет IBM PC для которых она выбрала 8-разрядные микросхемы 8088. В первом IBM PC использовался процессор 8088 с тактовой частотой 4,77 МГц, а на выполнение каждой команды процессор затрачивал примерно 12 тактов. Позже появились варианты с частотой 8 и 10 МГц, но это уже древняя история:).

Процессоры 80186 и 80188

После выпуска процессоров 8088 и 8086 Intel начала разработку более производительного процессора. Первые процессоры **80186** и **80188** были не очень удачными. Эти процессоры весьма похожи на своих прародителей 8086 и 8088. Процессор 80186 (как и 8086) был полностью 16-разрядным, а 80188 (как и 8088) - компромиссный вариант с внешней 8-разрядной и внутренней 16-разрядной шинами. Различие же между процессорами в том, что в их корпус помимо самого процессора встроено еще 15-20 компонентов, а это позволило резко сократить и сложность, и, в конечном счете, стоимость материнской платы. Однако такие процессоры редко использовались в PC, в первую очередь их применяли в различной периферийной технике и контроллерах.

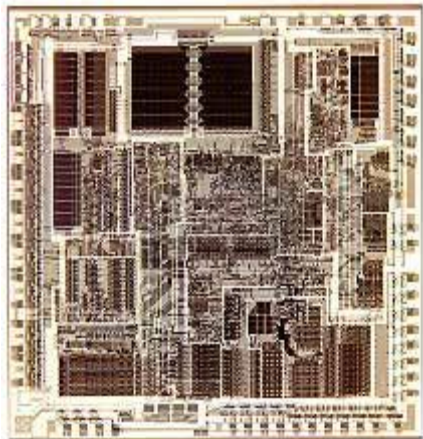
Сопроцессор 8087

Несколько позже процессора 8086 для него был разработан сопроцессор **8087**, который был предназначен для математических вычислений (операций с плавающей точкой). Такие

микросхемы, как вы помните из предыдущего урока, называли **FPU** (Floating-Point Unit). Также такой сопроцессор, как мы уже упоминали, мог производить сложные математические вычисления значительно быстрее, чем основной процессор.

3.6.2. Второе поколение процессоров.

Процессор 80286



Процессор 80286 (или просто **286**) был создан в 1981 году, и на его основе IBM создала компьютер, называемый IBM PC AT. Процессор имел 16-разрядные внутренние регистры и внешнюю шину данных, а так же мог адресовать 16 Мбайт оперативной памяти, так как его адресная шина составляла 24 бита. Производительность компьютера с процессором 286 с тактовой частотой 6 МГц была в пять раз выше, чем у системы с 4,77 МГц 8088. Это объясняется и возросшей тактовой частотой, и внешней шиной данных, шириной 16 бит и тем, что этот процессор исполнял одну команду в среднем за 4,5 такта (8088 - за 12). Кроме того, существовали версии 286 процессора с частотами работы от 6 до 20 МГц. Но даже при равных частотах (существует 8086 и 80286 на частоте 8 МГц) производительность нового

процессора была втрое выше. Ещё одним новшеством, которым порадовал мир этот процессор, явился обсуждаемый ранее «защищённый режим». Хотя, как вы помните из предыдущего урока, его реализация на этих процессорах не была удачной.

Сопроцессор 80287

Архитектура сопроцессора 80287 практически идентична архитектуре 8087. Несмотря на то, что 286 процессор работает на частоте, значительно превышающей 8088, сопроцессор 287 и по частоте практически не выигрывает у 8087. Можно сказать, что эффективность сопроцессоров 8087 и 80287 примерно равна, поэтому от установки в 286 систему сопроцессора, выигрыш оказывался совсем не столь значительным, как от установки сопроцессора в 8088 систему.

3.6.3. Третье поколение процессоров.

Процессор 386



Существует несколько разновидностей 386-го процессора:

Процессор 386DX

Процессор 386DX была первым процессором этого семейства. Он представлял собой полностью 32-битный процессор, у которого внутренние регистры, внутренняя и внешняя шина данных были 32-разрядными. Процессор мог адресовать 4 Гбайт оперативной памяти. Тактовая частота процессоров Intel 386DX колебалась от 16 МГц, до 33 МГц, а AMD выпускали ещё и 40 МГц версию этого процессора.

Процессор 386SX

Процессор 386SX предназначен для компьютеров с возможностями 386-го, который стоил бы не больше 286-ой системы. Он оказался урезанным вариантом 386DX – у него внутренняя шина данных и регистры были 32-разрядными, а внешняя шина – 16-разрядной.

Процессоры 386SX выпускались с тактовыми частотами 16...33 МГц.

Сопроцессор 387

В отличие от 287, который, по сути, являлся 8087 с другой разводкой ножек, сопроцессор 387 с повышенной производительностью разрабатывался специально для 386 систем. Intel не успела вовремя выпустить 387, первые платы для этих систем имели гнездо для подключения 287, а это производительности не добавляло :). Только более поздние платы для 386-го процессора имели гнездо для подключения 387-го сопроцессора. Существовало две разновидности 387: 387DX и 387SX, предназначенных соответственно для 386DX и 386SX процессоров.

3.6.4. Четвертое поколение процессоров.



Появление процессора 80486 (или просто 486) стало следующим этапом повышения быстродействия компьютеров. Все модификации процессора 486, как и 386DX являются полностью 32-разрядными, т.е. имеют 32-разрядные регистры, 32-разрядную внешнюю шину данных и адресную шину.

Эти процессоры стали первыми процессорами, в которых начали применять конвейерное исполнение команд, начали встраивать в ядро кэш-память первого уровня (L1) и математический сопроцессор. Существует несколько модификаций 486 процессора – 486DX, 486SX, 486DX2, 486SX2, 486DX4 и 486SX4. Рассмотрим их особенности:

Процессор 486 DX

Первый процессор 486DX был выпущен фирмой Intel в апреле 1989 года, а первый компьютер на его основе появился в 1990 году. Тактовая частота процессора составляла 25 МГц, затем появились модификации, рассчитанные на 33 и 50 МГц. Этот процессор являлся базовым 486-ым, имел встроенный сопроцессор и кэш-память первого уровня, емкостью 8 кб, а также контроллер этой памяти (кэш-контроллер).

Процессор 486 SX

Этот процессор начали выпускать в апреле 1991 года как более дешевый вариант CPU без сопроцессора. Ситуация с процессором 486SX отличается от 386-х процессоров: это полноценный 32-разрядный процессор, эквивалентный 486DX во всем, кроме отсутствия у 486SX сопроцессора. И в отличие от ситуации с 386DX и 386SX, обе разновидности 486-х процессоров (с сопроцессором и без него) являются взаимозаменяемыми, т.е. могут быть установлены в одну и ту же материнскую плату.

Процессоры 486SX появились скорее по коммерческим, а не по технологическим причинам. Дело в том, что процессоры 486SX представляли собой 486DX со сбойным сопроцессором, и при продаже процессор шел после того, как сопроцессор отключали.



Процессоры 486DX2, 486SX2 и 486DX4

В мае 1992 года Intel приступил к выпуску новых версий 486-ых процессоров, работающих на удвоенных, как в случае с DX2 и SX2, или утроенных как в случае с DX4 частотах. Эти процессоры с частотами 40, 50, 66, 80, 100, 120 МГц предназначались для замены младших версий 486 процессоров. Увеличение тактовой частоты в этих процессорах достигалось при помощи применения множителей, которые мы рассматривали в

прошлом уроке. Получается, что именно эти процессоры впервые применили множитель для получения частоты работы большей, чем частота системной шины.

3.6.5. Пятое поколение процессоров.

В пятом поколении выпускались следующие процессоры:

Pentium, или P5. Эти процессоры выпускались с тактовыми частотами 60 и 66 МГц, которые являлись частотой системной шины, и основными их особенностями были:

- Кэш-память первого уровня разделена на 2 части, отдельно по 8 Кб для данных и отдельно для команд, иногда обозначается как 8+8.
- Впервые применена суперскалярность.
- Шина данных имеет разрядность 64 бита, хотя процессор имеет 32-битные регистры.
- Модификаций без встроенного сопроцессора не выпускается, хотя именно из-за ошибки в нём эти процессоры и были сняты с производства.
- Для первого выпуска Pentium применялся процессорный разъём Socket 4. Технология изготовления этих процессоров была 0,8 мкм, и поэтому они очень сильно грелись, что тоже не прибавило им популярности.

Pentium, или P54. Этот процессор представлял собой слегка улучшенный вариант предшественника, и его единственное отличие составляет новый техпроцесс – 0,5 мкм. Для второго выпуска Pentium применялся сначала процессорный разъём Socket 5, позже - Socket 7. Частоты на которых работали – 75...200 МГц, причем частоты системной шины применялись 50, 60 и 66 МГц.

Аналогом этих процессоров были процессоры фирмы AMD, которые назывались **K5**. Технология изготовления была идентична – 0,5 мкм, а вот кэш L1 был немного увеличен – 16 Кб для команд, 8 Кб для данных. Встроенный сопроцессор серьёзно отставал от сопроцессора P54, и поэтому этот процессор составлял конкуренцию только в офисных приложениях.

Pentium MMX, или P55. Третий, и последний выпуск Pentium несет в своём имени название технологии, которая была в нём впервые представлена – MMX. Изготавливался он уже по технологии 0,35 мкм, имел вдвое больший кэш L1 – 32 Кб, по 16 на данные и команды. Устанавливался в Socket 7, и имел частоту системной шины только 66 МГц. При этом частоты с которыми они работали были 166, 200 и 233 МГц, ещё некоторое время выпускалась модификация для мобильных компьютеров, с частотой 266 МГц.

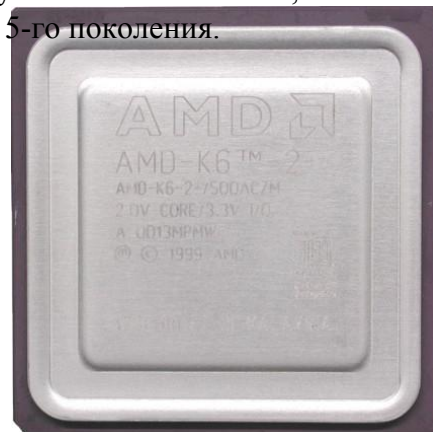


Конкуренцию этому процессору составлял процессор фирмы AMD, который назывался **K6**. До моделей с тактовой частотой 233 МГц, единственное его отличие в характеристиках от P55, был лишь вдвое больший кэш L1 – 64 Кб (32+32). В более поздних моделях, с тактовыми частотами 266 и 300 МГц, появилось ещё одно отличие – новый 0,25 мкм техпроцесс. Встроенный сопроцессор унаследовал от предыдущего процессора AMD низкую, по сравнению с Intel производительность.

Позже появилась модификация процессора K6 – **K6-2**, его основным отличием были наличие набора команд 3DNow!, этим набором команд AMD попытались заменить x87, который их сопроцессоры выполняли хуже, чем сопроцессоры Intel. Изготавливался этот процессор с

техпроцессом 0,25 мкм, и выпускался с тактовыми частотами 233...550 МГц, при этом частоты системной шины составляли 66 и 100 МГц. Следует учесть, что ввиду применения нового техпроцесса и частоты системной шины, для них требовались модифицированные разъёмы Socket 7, которые назывались Super 7. Этот процессор противопоставлялся уже не Pentium MMX, а Pentium II, хотя устанавливался на материнские платы для процессоров 5-го поколения.

На этом процессоре AMD не останавливаются, и поэтому выпускают модифицированный вариант K6-2, процессор **K6-III**. Он отличается от предшественника наличием встроенного в кристалл процессора кэша L2, объёмом 256 Кб, который работает на частоте процессора. В результате процессор хоть и получился более производительным, его цена была очень высока. Конкурировал он уже с процессорами Pentium III, но из-за морального устаревания платформы и проблем с выпуском, серьёзной конкуренции составить не мог.



Последними процессорами в семействе K6, являются процессоры **K6-2+** и **K6-III+**, оба этих процессора выполнены с техпроцессом 0,18 мкм, и оба имеют кэш L2 встроенный в ядро - K6-2+ - 128 Кб, K6-III+ - 256 Кб.

3.6.6. Процессоры шестого поколения.

Pentium Pro

Тактовые частоты - от 150 до 200 МГц.
Частоты системной шины - 60 и 66 МГц.
Процессорный разъем - Socket 8.
Кэш-память L1 - 16 кбайт.
Кэш-память L2 - 256/512/1024/2048 кбайт.
Технология изготовления - 0,5 мкм, затем - 0,35 мкм.
Поддерживаемые наборы команд – x86, x87



Первым процессором шестого поколения стал процессор Pentium Pro. Внутри корпуса процессора на самом деле содержалось две различных микросхемы: собственно ядро процессора и кэш памяти второго уровня. Основной особенностью этого процессора является двойная независимая шина (DIB), которая состоит из 64-битной шины кэш-памяти (на частоте процессора) и внешней шины данных процессора.

Изготовление процессоров Pentium Pro было крайне дорого из-за того, что в одном корпусе нужно было разместить два различных кристалла, поэтому цена даже на младшие модели с 256кб кэша L2 была весьма высокой. Из-за проблем с производством и, как следствие, высокой цены, эти процессоры никогда не стали процессором настоящего PC, так навсегда и оставшись СУГУБО СЕРВЕРНЫМ решением.

Pentium II Klamath

Тактовые частоты – 233...300 МГц.
Частота системной шины (FSB) - 66 МГц.
Процессорный разъем - Slot 1.
Кэш-память L1 - 32 Кб, (16+16).
Кэш-память L2 - 512 Кб.



Технология изготовления - 0,35 мкм.

Поддерживаемые наборы команд – x86, x87, MMX

Главное изменение в этом процессоре коснулось кэша второго уровня. Intel отказывается от такого дорогостоящего, как в Pentium Pro, способа изготовления процессора с кэшем, и применяет другой подход. Процессор Pentium II представляет собой только кристалл процессора, без кэша L2 внутри, и этот кристалл припаян на небольшого размера картридж, на котором помимо самого процессора, установлен и кэш второго уровня. Обмен кэш-процессор происходит уже не на полной частоте процессора, как у Pentium Pro, а на половине частоты процессора. Это сделано для того, чтобы уменьшить стоимость процессора, так как полноскоростной кэш, устанавливаемый в картридж Pentium II, сделал бы процессор очень дорогим.

Pentium II Deschutes

Тактовые частоты – 266...450 МГц.

Частоты FSB – 66, 100 МГц.

Процессорный разъем - Slot 1.

Кэш-память L1 - 32 Кб.

Кэш-память L2 - 512 Кб.

Технология изготовления - 0,25 мкм.

Поддерживаемые наборы команд – x86, x87, MMX

Единственным отличием этого процессорного ядра от предшественника, является более совершенный техпроцесс – 0,25 мкм, и как следствие, возможность использования больших частот процессора и системной шины.



Celeron Covington

Тактовые частоты - 266 и 300 МГц.

Частота системной шины - 66 МГц.

Процессорный разъем - Slot 1.

Кэш-память L1 - 32 кбайта.

Кэш-память L2 - 0 (!) кбайт.

Технология изготовления - 0,25 мкм.

Поддерживаемые наборы команд – x86, x87, MMX

В то время как Intel начала выпускать процессоры 6-го поколения, AMD активно взялись осваивать рынок Socket 7 систем. процессоры семейства K6 стоили намного дешевле Pentium II, и по этой причине составляли им серьезную конкуренцию. Поэтому Intel выпускает удешевлённый Pentium II – Celeron. По сути этот процессор представляет из себя то же ядро Deschutes, но без КЭШа L2.

Процессор стоил недорого, но был крайне несбалансированным: быстрое процессорное ядро, хороший сопроцессор, но очень медленная подсистема памяти из-за отсутствия кэша второго уровня. Процессор Celeron Covington был бы совсем непопулярен у пользователей, если бы не одно обстоятельство. Благодаря отсутствию кэша L2, процессор прекрасно разгонялся, например с 300 МГц до 450 МГц, чем и привлекал многих пользователей.



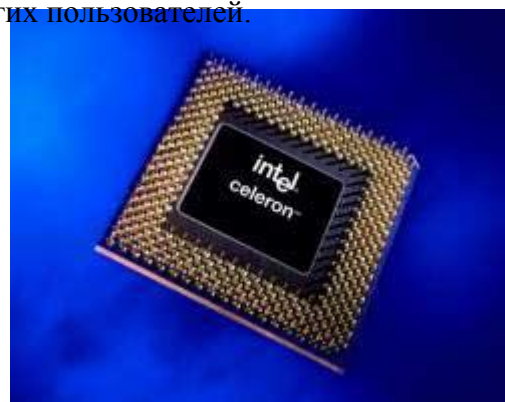
Celeron Mendocino

Тактовые частоты - 300-533 МГц.

Частоты FSB - 66 МГц.

Процессорный разъем - Slot 1, затем Socket 370.

Кэш-память L1 - 32 Кб.



Кэш-память L2 - 128 Кб.

Технология изготовления - 0,25 мкм.

Поддерживаемые наборы команд – x86, x87, MMX

В отличие от Covington, новый Celeron был оборудован кэшем второго уровня, размером 128 Кб, причем он работал на частоте процессора. Стоимость такого процессора вместе с кэшем была лишь немного выше, чем процессора без кэша, и для избегания путаницы с маркировкой процессоров работающих на одинаковой частоте, к ней (частоте) добавили индекс "А". Например, Covington 300 МГц - Celeron 300, а Mendocino 300 МГц - Celeron 300A.

Pentium !!! Katmai

Тактовые частоты – 450...600 МГц.

Частота FSB - 100 МГц.

Процессорный разъем - Slot 1.

Кэш-память L1 - 32 Кб.

Кэш-память L2 - 512 Кб.

Технология изготовления - 0,25 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE

Этот процессор отличался от Deschutes только одним: новый процессор поддерживал новый набор команд, который до выхода процессора называли KNI (Katmai New Instruction), а официальное название нового набора команд – SSE.

Pentium !!! Coppermine

Тактовые частоты – 500...1133 МГц.

Частоты FSB - 100 и 133 МГц.

Процессорный разъем – Slot 1, Socket 370.

Кэш-память L1 - 32 Кб.

Кэш-память L2 - 256 Кб.

Технология изготовления - 0,18 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE

В этом варианте Pentium III изменился кэш второго уровня. Его стало вдвое меньше - 256кб, но при этом кэш L2 располагался не на процессорной платке, а непосредственно в кристалле процессора, как и у Mendocino. За счет этого скорость работы кэша второго уровня равнялась скорости работы процессорного ядра. За счет нового кэша ядро Coppermine было несколько быстрее ядра Katmai. Также новая технология производства – 0,18 мкм, дала возможность использовать более высокие частоты.

Celeron Coppermine-128

Тактовые частоты - 533-766 и 800-1100 МГц.

Частоты FSB - 66 и 100 МГц.

Процессорный разъем - Socket 370.

Кэш-память L1 - 32 кбайта.

Кэш-память L2 - 128 кбайт.

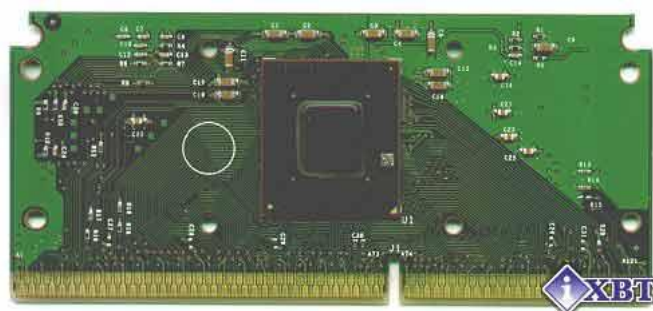
Технология изготовления - 0,18 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE

На базе нового процессорного ядра Coppermine Intel решила выпустить и процессор класса Celeron, ядро этого нового процессора получило название Coppermine-128.

Тактовые частоты в модельном ряде: 533, 566, 600, 663, 667,

© Вячеслав Калашников, Дмитрий Боровик, Руслан Диденко



700, 733, 766 МГц - при частоте системной шины 66 МГц; 800, 850, 900, 950, 1000, 1100 МГц - при частоте системной шины 100 МГц.

Pentium !!! Tualatin

Тактовые частоты – 933...1400 МГц.

Частоты FSB - 133 МГц.

Процессорный разъем - Socket 370.

Кэш-память L1 - 32 Кб.

Кэш-память L2 - 256 Кб или 512 Кб.

Технология изготовления - 0,13 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE



Интересный факт – это процессорное ядро появилось позже процессора Pentium 4, который мы рассмотрим позже, и благодаря техпроцессу 0,13 мкм получило очень неплохой потенциал к разгону. В результате процессор оказался в некоторых приложениях лучше, чем Pentium 4. В результате, Intel прекратили выпуск настольных вариантов этих процессоров, оставив их только в серверных и мобильных модификациях.

Celeron Tualatin

Тактовые частоты – 1000...1400 МГц.

Частоты FSB - 100 МГц.

Процессорный разъем - Socket 370.

Кэш-память L1 - 32 Кб.

Кэш-память L2 - 256 Кб.

Технология изготовления - 0,13 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE

Стоит заметить, что этот Celeron отличается от полноценного Pentium III только частотой системной шины. А 0,13 мкм техпроцесс позволил легко разгонять эти процессоры, и получать полноценный Pentium III по цене Celeron. Надо сказать, что для своего времени, именно этот Celeron явился самым удачным среди своих собратьев, т.к. отличий от Pentium III совсем немного:).

Теперь стоит уделить внимание процессорам 6-го поколения, которые выпускались основным конкурентом Intel – фирмой AMD.

Процессор AMD Athlon (Argon)

Тактовые частоты – 500...1000 МГц.

Частоты FSB - 2x100 (DDR) МГц.

Процессорный разъем - Slot A.

Кэш-память L1 - 128 Кб.

Кэш-память L2 - 512 Кб.

Технология изготовления - 0,25 (затем 0,18) мкм.

Поддерживаемые наборы команд – x86, x87, MMX, 3DNow!



Новую серию процессоров AMD называли K7, как продолжение семейства K5 и K6.
© Вячеслав Калашников, Дмитрий Боровик, Руслан Диденко

Основная особенность этого процессора – системная шина. Если в предыдущих процессорах AMD старались сохранить совместимость с системной шиной процессоров Intel, то в этих процессорах они применили системную шину, которая не была совместима с шинами Intel. И это видно из приведенных характеристик – она работает в режиме DDR, т.е. передаёт данные с вдвое большей частотой, чем частота сигнала в шине. Самое интересное – процессорный разъём Slot A, который идентичен перевёрнутому Slot 1, но не абсолютно с ним не совместим. AMD решили облегчить жизнь производителям материнских плат возможностью применения одного и того же разъёма для разных процессоров.

Также в процессоре Athlon был серьёзно переработан блок FPU – он уже не отстаёт от Intel, а по заявлениям AMD даже обгоняет его.

Процессор AMD Athlon (Thunderbird)

Тактовые частоты – 650...1400 МГц.

Частоты FSB - 2x100 (DDR) и 2x133 (DDR) МГц.

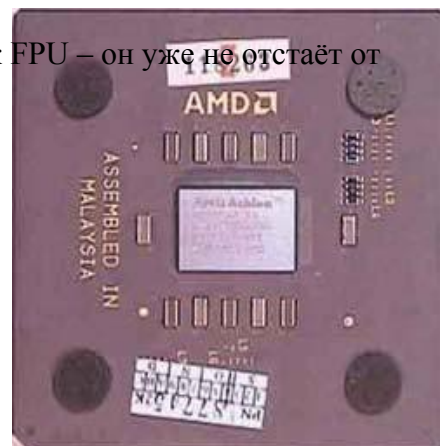
Процессорный разъём – Slot A, Socket A (Socket 462).

Кэш-память L1 - 128 кбайт.

Кэш-память L2 - 256 кбайт.

Технология изготовления - 0,18 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, 3DNow!



Процессор Thunderbird в отличие от Argon имеет уже 256 Кб кэша второго уровня, встроенного в ядро, и работающего на полной тактовой частоте процессора. С этого процессора также началось применение нового процессорного разъёма Socket A. На этом отличия заканчиваются.

Процессор AMD Duron (Spitfire)

Тактовые частоты – 600...950 МГц.

Частоты системной шины - 2x100 (DDR) МГц.

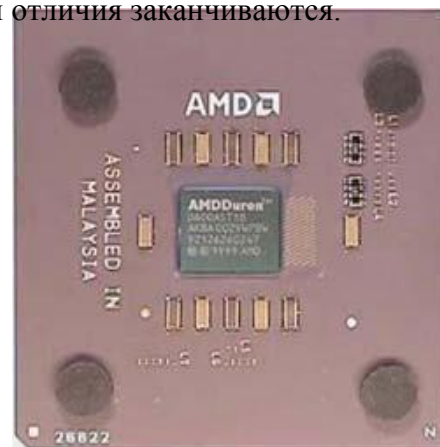
Процессорный разъём - Socket A (Socket 462).

Кэш-память L1 - 128 Кб.

Кэш-память L2 - 64 Кб.

Технология изготовления - 0,18 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, 3DNow!



Вслед за Intel Celeron, AMD тоже решает выпустить удешевлённый вариант процессора Athlon – процессор Duron, который и будет противостоять Celeron на рынке недорогих процессоров.

Этот процессор отличался от Athlon только лишь уменьшенным кэшем второго уровня: он составлял не 256 кб, а 64 кб. Получается, что у Duron кэша вдвое меньше, чем у Celeron? Нет! Дело в том, что у процессоров Intel так называемый включительный кэш, т.е. содержимое кэша первого уровня всегда присутствует и в кэше второго уровня. Таким образом у Pentium III и Celeron суммарный объём кэша равен количеству кэша второго уровня, а у процессоров AMD кэш эксклюзивный, т.е. содержимое L1 не дублируется в L2, следовательно, всего кэша у Duron даже больше чем у Celeron. А учитывая разницу в производительности FSB, получается что Duron выигрывает у Celeron по производительности. Мало того, процессор Duron стоил гораздо дешевле чем Celeron на равной тактовой частоте.

3.7. Современные процессоры

3.7.1. Процессоры 7-го поколения Intel.

Pentium 4 Willamette

Тактовые частоты – 1200...2000 МГц.

Частота FSB - 400 (4x100) МГц.

Процессорный разъем - Socket 423, Socket 478.

Кэш-память L1 - 8 Кб данные, 12000 микрокоманды.

Кэш-память L2 - 256 кбайт.

Технология изготовления - 0,18 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2.



Начнем с рассмотрения физического устройства данного процессорного ядра. Pentium 4 Willamette изготавливался по 0,18 мкм технологии. Этот процессор показывает новую архитектуру – **NetBurst**. Её основные составляющие, это **trace cache**, **quad pumped** шина данных процессора, **Hyper Pipelined Technology** – удлинённый конвейер, **Advanced Dynamic Execution** – улучшенное предсказание переходов и исполнение команд с изменением порядка их следования, и наконец **Rapid Execute Engine** – АЛУ, работающее на вдвое большей частоте чем весь процессор.

Немного подробнее об этих особенностях.

Кэш L1 для данных сократился до 8 Кб, против 16 в Pentium III. Кэш L1 для команд в привычном понимании отсутствует, его заменил **trace cache**. В нем хранятся последовательности микроопераций, в которые декодированы инструкции. В этом кэше могут помещаться до 12К микроинструкций.

Шина данных этого процессора работает с «четырёхкратной накачкой», **quad pumped**. Т.е. при частоте работы шины 100 МГц, частота передачи данных составляет 400 МГц.

Разрядность шины данных, как и в предыдущих двух поколениях процессоров, составляет 64 бита (8 байт), что в режиме 4х-кратной передачи дает максимальную пропускную способность 100x4x8 - 3,2 Гб/с. У процессоров Pentium III шина обеспечивала 133x8 - 1,06 Гб/с, так что по этому параметру у Pentium 4 улучшение втрое. Шина адреса имеет разрядность 36 бит, что позволяет адресовать те же 64 Гбайт памяти, из которых кэшируются только первые 4 Гбайт.

Этот процессор позиционировался как высокопроизводительное настольное решение. Основным отличием новой архитектуры от "старушки" P6 было еще большее увеличение конвейера — до 20 стадий, что позволило сильно нарастить частоту процессора без перехода на меньший техпроцесс. Дело в том, что, чем длиннее конвейер, тем легче наращивать тактовую частоту, но тем меньше производительности получается на каждый полученный мегагерц. И наоборот. Это происходит потому, что чем на большее количество стадий рассчитан конвейер, тем меньше работы приходится на каждый отдельный такт, тем быстрее этот самый такт выполняется. Willamette по сравнению с современными ему процессорами является абсолютным чемпионом по длине конвейера, то есть, имеет самое меньшее время выполнения такта, позволяющее достичь максимальной тактовой частоты, правда, возможны и самые большие задержки для связанных друг с другом операций. Да, и еще одна маленькая особенность - декодер команд не входит в 20 стадий конвейера NetBurst.

Интересный факт - арифметико-логическое устройство данных процессоров работает на буквально "заоблачных" частотах — они в два раза больше по сравнению с частотой ядра. Хотя прирост производительности получился всего до 60%, и то на некоторых

операциях.

Ещё хочется отметить, что в ядре этого процессора присутствует температурный датчик, который при первых признаках перегрева понизит частоту процессора, и не даст ему выйти из строя при поломке системы охлаждения. Надо сказать, что до этого такие датчики располагались на материнских платах, и их эффективность была невысокой, т.к. между процессором и датчиком оставалась воздушная прослойка, которая мешала качественному теплообмену. Несмотря на всё это, результаты первых тестов новых системнапрочь испортили им репутацию. По производительности Pentium 4 нередко уступал привычным, проверенным и относительно недорогим Pentium III Coppermine, не говоря уж о более поздних Tualatin'ax. Надо ли говорить, что завоевывать популярность таким дорогим и, мягко говоря, не совсем удачным процессорам пришлось бы очень долго. Этим и воспользовались AMD в своих новых процессорах Athlon XP, но об этом позже.

Celeron Willamette

Тактовые частоты - 1700-1900 МГц.

Частота FSB - 400 (4x100) МГц.

Процессорный разъем - Socket 478.

Кэш-память L1 - 8 Кб, 12К микрокоманд.

Кэш-память L2 - 128 кбайт.

Технология изготовления - 0,18 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2.



Новая линейка процессоров Celeron, была основана на ядре Willamette с урезанным (по сравнению с оригинальным Willamette — Pentium 4) до 128 КБ объемом кэш-памяти второго уровня и просто и незатейливо называемом «Willamette-128».

Рассматривая ранее процессоры семейства Pentium 4, основанные на NetBurst архитектуре, мы отметили тот факт, что по своей производительности они нередко уступают процессорам с аналогичной или меньшей частотой, но иной архитектурой. Поэтому перевод Celeron на архитектуру NetBurst вызвал у многих компьютерных энтузиастов вполне естественные размышления относительно производительности таких процессоров. Иногда даже высказывались мнения, что новые Celeron с частотами 1.7 и 1.8 ГГц будут работать медленнее своих предшественников с частотами 1.3 и 1.4 ГГц, использующих ядро Tualatin, и иногда так и было.

Pentium 4 Northwood

Тактовые частоты – 1600...3200 МГц.

Частоты FSB - 400 (4x100), 533 (4x133), 800 (4x200) МГц.

Процессорный разъем - Socket 478.

Кэш-память L1 - 8 Кб, 12000 микрокоманд.

Кэш-память L2 - 512 Кб.

Технология изготовления - 0,13 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2



Процессоры Pentium 4, основанные на новом 0,13 мкм ядре Northwood имеют в названии индекс А, для избежания путаницы с процессорами на ядре Willamette работающими на тех же частотах. Например, это Pentium 4 2А - первая модель на Northwood'e, работающая на частоте 2 ГГц. Любопытно отметить, что процессор на новом ядре содержит 55 миллионов транзисторов (площадь — 146 кв. мм), из которых около 40% (у Willamette 42 миллиона транзисторов, а изменился только объем L2) приходится на кэш, а ведь это немало.

Что касается FSB, то тут Pentium 4 Northwood оказался на высоте, осилив следующие ступени - 533 и 800 МГц, что не могло не сказаться на его производительности - процессоры с частотой Front-Side Bus 533 МГц, например, оказались в среднем на 5-10% быстрее своих аналогов с FSB 400 МГц. Самое интересное - цены на новые процессоры Intel выставила примерно равные ценам на процессоры с предыдущим значением FSB (разница примерно 1 USD), так что линейка старых моделей Willamette, должна была просто сойти со сцены по мере появления все новых CPU. Модели с FSB 533 МГц в своей маркировке имели индекс В, а с FSB 800 МГц индекс С. Получается, что процессор Pentium 4 с частотой 2,4 ГГц существовал в 3-х вариантах - Pentium 4 2,4А, Pentium 4 2,4В и Pentium 4 2,4С. Соответственно FSB у первого – 400 МГц, у второго – 533 МГц, а у третьего – 800 МГц. Получается один и тот же процессор, но с разной пропускной способностью системной шины.

Ещё одно новшество - Hyper-Threading, который присутствовал в некоторых моделях с FSB 533 МГц, и во всех моделях с FSB 800 МГц.

В общем, новый процессор от Intel получился очень неплохим не только в кремнии, но и в тестах показал себя хорошо, хотя особых революционных нововведений он не продемонстрировал (ведь увеличение кэша L2 и уменьшение техпроцесса до 0,13 мкм не в счет - тем более, что такой техпроцесс уже применялся в Tualatin).

Celeron Northwood

Тактовые частоты – 2000...2800 МГц.

Частоты FSB - 400 (4x100)

Процессорный разъем - Socket 478.

Кэш-память L1 - 8 Кб, 12000 микрокоманд.

Кэш-память L2 - 128 Кб.

Технология изготовления - 0,13 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2



От этого процессора ожидали большего, чем он явился на самом деле. Ожидалось что с появлением процессоров Pentium 4 Northwood, Intel для соответствующего ему Celeron'a оставит половину кэша L2 относительно нормального процессора, как было в случае с Tualatin, однако этот процессор вышел с вчетверо меньшим кэшем, что огорчило многих пользователей. Однако не всё так плохо как кажется, плюсом этого процессора оказалось то, что он очень неплохо разгоняется. Например 2 ГГц модель при установке FSB 533 МГц, во многих случаях без затруднений становилась 2,66 ГГц:).

Pentium 4 Prescott

Тактовые частоты – 2,26...3,8 ГГц.

Частоты FSB - 533 (4x133), 800 (4x200) МГц.

Процессорный разъем - Socket 478, LLGA 775.

Кэш-память L1 - 16 Кб, 12000 микрокоманд.

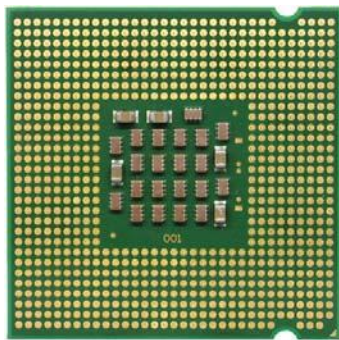
Кэш-память L2 - 1024 Кб.

Технология изготовления - 0,09 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2, SSE3.



Изначально эти процессоры выпускались для



использовавшегося ранее Socket 478, но позже появились процессоры

для принципиально нового LLGA 775. На фотографии изображен

такой процессор. Что же в новом разъёме? Дело в том, что процессор

для такого разъёма не имеет ножек, их заменили контактные

площадки, а ножки располагаются в процессорном разъёме, и

процессор прижимается к ним специальной рамкой. В результате Intel

обеспечили лучший контакт, чем можно было обеспечить массивом

ножек на процессоре.

Стоит заметить, что этот процессор имеет вдвое большие кэши

L1 и L2. Кроме этого в нём усовершенствован Hyper-Threading, и несколько доработан

механизм предсказания переходов. Этого и стоило ожидать – длина конвейера увеличена до

31 стадии, и производительность стала ещё больше зависеть от механизма предсказания переходов.

Процессоры выпускаемые для Socket 478 отличить от предшественника, Pentium 4

Northwood, можно только по кэшу L2. А для процессоров на LLGA 775 Intel придумали новую

рейтинговую систему обозначений. В этой системе процессоры обозначаются тремя цифрами,

первая из которых обозначает класс процессора. В настоящее время есть 4 класса процессоров: серия 800 – высокопроизводительные двухядерные Pentium D, серия 700 –

процессоры для портативных компьютеров Pentium M, 600 и 500 серии – настольные варианты Pentium 4, отличающиеся друг от друга только объёмом кэша L2, в 600-й серии – 2 Мб, в 500-й серии 1 Мб. И последняя – 300-я серия, удешевлённые Celeron D. Давайте чуть подробнее

рассмотри эти модификации.

Pentium D 8xx

Тактовые частоты – 2,8...3,2 ГГц.

Частоты FSB - 800 (4x200) МГц.

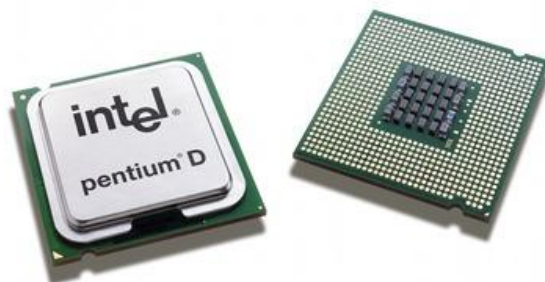
Процессорный разъем - LLGA 775.

Кэш-память L1 - 16 Кб, 12000 микрокоманд.

Кэш-память L2 – 2x1024 Кб.

Технология изготовления - 0,09 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2, SSE3.



Основное отличие этой серии процессоров – использование двух процессорных ядер, вместо одного, фактически этот процессор представляет сдвоенное ядро Prescott, и у каждого ядра присутствует по 1 Мб кэша L2, но они при этом используют одно пространство памяти. Если вы помните из предыдущего урока, такая архитектура называется CMP. Ещё эти процессоры поддерживают рассматриваемую в прошлом уроке технологию NX-bit, которая якобы представляет из себя аппаратную защиту от вирусов. Кроме этого процессоры 800-й серии не поддерживают Hyper-Threading, но включают в себя технологию EM64T, проще говоря – являются 64-разрядными.

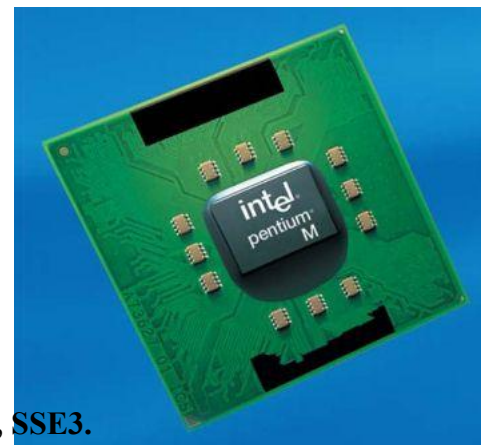
Pentium M 7xx

Тактовые частоты – 1...2,26 ГГц.
Частоты FSB - 400 (4x100), 533 (4x133) МГц.
Процессорный разъем – Socket 479.
Кэш-память L1 – 64 Кб (32+32).
Кэш-память L2 – 512 Кб, 1024 Кб.
Технология изготовления - 0,09 мкм.
Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2.

При детальном рассмотрении этого процессора, оказывается, что это доработанный Pentium III, т.е. его перевели на новый техпроцесс, увеличили кэш, причем как L1 так и L2, и адаптировали его к FSB Pentium 4. Получился неплохой мобильный процессор, причем его можно применять и в качестве настольного процессора – существуют переходники Socket 479 - Socket 478.

Pentium 4 6xx

Тактовые частоты – 3...3,8 ГГц.
Частоты FSB - 800 (4x200) МГц.
Процессорный разъем – LGA 775.
Кэш-память L1 - 16 Кб, 12000 микрокоманд.
Кэш-память L2 – 2048 Кб.
Технология изготовления - 0,09 мкм.
Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2, SSE3.



Этот процессор представляет усиленное ядро Prescott, названное Prescott 2M. В этом ядре присутствует 2 Мб кэша L2. Также в них есть поддержка технологии Hyper Threading (HT), EM64T и NX-bit.

Pentium 4 5xx

Тактовые частоты – 2,66...3,8 ГГц.

Частоты FSB - 800 (4x200), 533 (4x133) МГц.

Процессорный разъем – LLGA 775.

Кэш-память L1 - 16 Кб, 12000 микрокоманд.

Кэш-память L2 – 1024 Кб.

Технология изготовления - 0,09 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2, SSE3.

Этот процессор – просто Prescott, но выполненный в корпусе для разъёма LLGA 775.

Причем модификаций этого процессора много – есть модели на 533 FSB, вторая цифра их номера 0 или 1, например 505 или 516, эти модели выпускаются без HT. Есть модели без EM64T и NX-bit, их индекс заканчивается просто на 0 или 5, например 540 или 505. Есть модели в которых нет только EM64T, а HT и NX-bit присутствуют, это номера с индексом J, например 540J. И напоследок – модели, которые отличаются от 600-й серии только размером кэша, заканчиваются на 1, например 571.

Celeron D 3xx

Тактовые частоты – 2,26...3,2 ГГц.

Частоты FSB - 533 (4x133) МГц.

Процессорный разъем – LLGA 775, Socket 478.

Кэш-память L1 - 16 Кб, 12000 микрокоманд.

Кэш-память L2 – 256 Кб.

Технология изготовления - 0,09 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2, SSE3.

Эти процессоры представляют из себя урезанный вариант Prescott, причем в отличии от предыдущего Celeron, они работают на частоте FSB 533 МГц, и имеют кэш L2 256 Кб. Причем некоторые модели так и выпускаются для старого Socket 478, а большая часть этих процессоров для LLGA 775. Hyper-Treading они не поддерживают, однако некоторые модели для LLGA 775 имеют поддержку EM64T и NX-bit.

3.7.2. Процессоры 7-го поколения AMD.

Athlon XP Palomino

Тактовые частоты – 1333...1733 МГц.

Частота FSB - 266 (2x133) МГц.

Процессорный разъем - Socket 462.

Кэш-память L1 - 128 Кб.

Кэш-память L2 - 256 Кб.

Технология изготовления - 0,18 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, 3DNow!



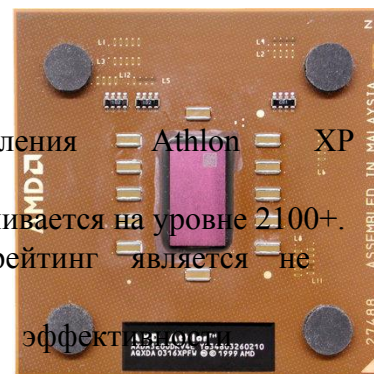
Чуть позже Pentium 4 на рынке появились процессоры AMD, выполненные по 0,18 мкм технологии. Palomino - кодовое наименование ядра процессоров Athlon, пришедшего на смену архитектуре Thunderbird. В Palomino имеются незначительные архитектурные изменения с целью улучшения скоростного потенциала процессора: в составе ядра улучшенный блок предсказания ветвлений и аппаратная предварительная выборка из памяти. Процессоры на новом ядре поддерживают усовершенствованный набор 3DNow! Professional, усовершенствование которого представляет собой добавление в него набора SSE:). Palomino имеет большую производительность, чем Thunderbird, работающий на той же частоте. Процессорный разъем для процессоров Athlon XP - Socket 462, тот же, что и для предыдущих процессоров.

В отличие от продукта своего ближайшего конкурента, первые модели Athlon XP вполне успешно могли работать и на старых Socket A-платах, лишь бы те поддерживали необходимую частоту. Как показала практика, в большинстве случаев удавалось даже обойтись без перепрошивки BIOS.

В этом ядре AMD вслед за Intel решили также встраивать датчик температуры, но в первых выпусках Athlon XP этот датчик сгорал раньше процессора, так что его эффективность была сомнительна.

Интересно, что AMD решила маркировать данные процессоры не их реальной частотой ядра, а так называемым PR-рейтингом (Pentium Rating - это показатель примерного соответствия производительности процессора в приложениях общего характера, например, распространенных ОС, типовых офисных программах, играх средней сложности, процессору Intel Pentium 4c указанной тактовой частотой). Он логически вписывается в

существующую линейку Athlon'ов, частоты которых, как мы сказали, до появления Athlon XP заканчивались отметкой 1,4 ГГц. PR Athlon XP начинается с отметки 1500+ и заканчивается на уровне 2100+. Вместе с тем, надо отдавать себе отчет в том, что PR-рейтинг является не более чем маркетинговым средством, направленным на увеличение эффекта продвижения продукта на рынке. Реально, PR указывает на частотный эквивалент приблизительно соизмеримого по производительности Pentium 4 процессора. С другой стороны, подобная оценка весьма условна, если принять во внимание существенные различия в архитектурах обоих процессоров. При одинаковой частоте Pentium 4 в мегагерцах с PR-рейтингом Athlon XP лидера определяет только приложение — в одних программах лучшие результаты по казывает новый Athlon, в других - Pentium.



Duron Morgan

Тактовые частоты - 900-1400 МГц.

Частоты FSB - 200 (2x100) МГц.

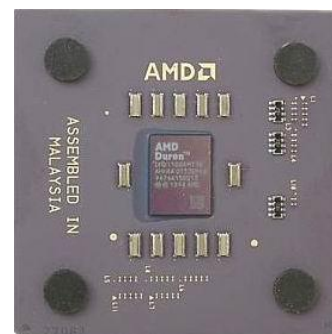
Процессорный разъем - Socket 462.

Кэш-память L1 - 128 кбайт.

Кэш-память L2 - 64 кбайт.

Технология изготовления - 0,18 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, 3DNow!



Фактически, ядро Morgan отличается от процессорного ядра Palomino, лишь размером кэша второго уровня. Процессоры на ядре Morgan оснащаются 64-килобайтным кэшем, в то время как Palomino располагает L2 кэшем с размером 256 Кбайт. Маркировку рейтингом для этого процессора AMD решили не применять. Вторым различием Morgan и Palomino, имеющим явно маркетинговые корни, является различная используемая ими частота системной шины. Процессоры Palomino работают с 266-мегагерцевой шиной с пропускной способностью 2.1 Гбайт/с, в то время как Morgan использует шину с частотой 200 МГц и пропускной способностью 1,6 Гбайт/с. Впрочем, это различие имеется только на бумаге: в архитектуре Morgan нет никаких ограничений, которые не позволили бы этому процессору работать и на 266-мегагерцевой шине.

Athlon XP Thoroughbred

Тактовые частоты – 1467...1733 МГц.

Частоты FSB - 266 (2x133) МГц.

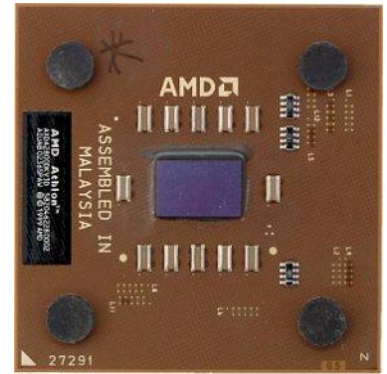
Процессорный разъем - Socket 462.

Кэш-память L1 - 128 кбайт.

Кэш-память L2 - 256 кбайт.

Технология изготовления - 0,13 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, 3DNow!



Единственным отличием этого ядра от предшественника, Palomino, является новая 0,13

мкм технология производства. Впрочем это типичный случай, когда ядро выполненное по

меньшей технологии меньше в размерах (сравните фото), потребляет меньшее напряжение и

имеет меньшее тепловыделение. Также благодаря меньшей технологии они стали идеальными процессорами для разгона – процессоры с рейтингом 1700+ отлично разгоняются до 2500+.

Чуть позже появляется переработанная версия этого ядра - Thoroughbred B,

представляющая из себя урезанный вариант ядра Barton, рассматриваемого далее.

Отличается Thoroughbred B только частотами – от 1467 МГц до 2250 МГц, причем есть модели

с FSB 333 (2x166) МГц.

Рейтинг первой версии ядра Thoroughbred составлял от 1700+ до 2200+, а Thoroughbred B – от 1700+ до 2800+. Получается что можно найти процессор Athlon XP с одним рейтингом,

но с разными ядрами, например популярный одно время Athlon XP 1700+ встречался на ядре Palomino, Thoroughbred и Thoroughbred B.

Athlon XP Barton

Тактовые частоты – 1,833...2,2 ГГц и выше.

Частоты FSB - 333 (2x166) МГц.

Процессорный разъем - Socket 462.

Кэш-память L1 - 128 Кб.

Кэш-память L2 - 512 Кб.

Технология изготовления - 0,13 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, 3DNow!

Barton – модификация ядра Thoroughbred с кэшем L2, увеличенным в два раза, и частотой FSB 333 МГц. Изготавливается Barton с применением улучшенной технологии SOI (Silicon-On-Insulator - Кремний-На-Изоляторе). Использование этой технологии позволяет увеличить тактовые частоты примерно на 20% и уменьшить при этом энергопотребление. Причем никаких особенных изменений нет, как хорошо видно на приведенных фотографиях: фактически, кристалл просто удлинен для размещения дополнительного кэша, а общая площадь ядра при этом возросла с 84 до 101 кв. мм. количество транзисторов — с 37,6 до 54,3 млн. штук.

Впрочем, есть надежда, что теперь, наконец, получит ход одна старая именно «глубинная» технология процессоров AMD, внедренная еще во времена Palomino, но, как до недавнего времени и встроенный термодатчик, реально не задействовавшаяся материнскими платами. Это довольно элегантное решение вечной для Athlon XP проблемы с тепловыделением давно известно разработчикам мобильных и встраиваемых(embedded) систем, и заключается оно в способности CPU при простое «отключаться» от системной шины (отсюда и название функции: System Bus Disconnect). Учитывая длину проводников, идущих от северного моста чипсета к процессору, и частоту, на которой работает шина Barton и последних Thoroughbred, любой начинающий радиоинженер вполне может себе представить, какую нагрузку по питанию создают буфера шины, расположенные в CPU. Поэтому решение дать им возможность иногда «отдохнуть» представляется вполне разумным.

Рейтинг этих процессоров составлял от 2500+ до 3200+.

Duron AppleBred

Тактовые частоты – 1,4...1,8 ГГц и выше.

Частоты FSB - 266 (2x133) МГц.

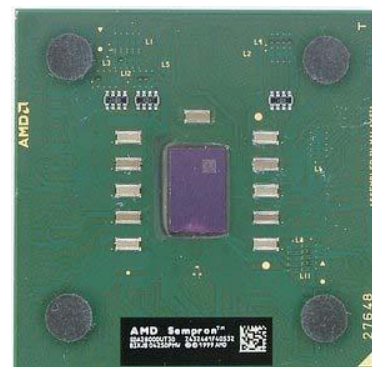
Процессорный разъем - Socket 462.

Кэш-память L1 - 128 Кб.

Кэш-память L2 - 64 Кб.

Технология изготовления - 0,13 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, 3DNow!



Выпуск этого процессора был воспринят как недоразумение – удешевлённый процессор с ценой на 10...15\$ меньшей чем Athlon XP, и при этом всего лишь с 64 Кб кэша L2 – далеко не самый лучший выбор для покупки. Хотя разработчиков этого процессора можно

понять – при

детальном рассмотрении оказывается что он представляет из себя процессор Thoroughbred В с бракованным кэшем L2. самое интересное, что некоторые умельцы научились при небольшой

модификации процессора "расширять" кэш L2 до 256 Кб. Хотя как говорилось – разница в цене

была незначительной, и толк от такого расширения тоже не небольшой – проще купить полноценный Athlon XP.

Sempron Thoroughbred

Тактовые частоты – 1,5...2,0 ГГц и выше.

Частоты FSB - 333 (2x166) МГц.

Процессорный разъем - Socket 462.

Кэш-память L1 - 128 Кб.

Кэш-память L2 - 256 Кб, 512 Кб (только на ядре Barton).

Технология изготовления - 0,13 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, 3DNow!

AMD с выпуском процессоров Athlon 64, решили что торговая марка Athlon будет использоваться для новых высокопроизводительных процессоров, а процессоры производимые

на ядрах для Athlon XP выпускать с новой торговой маркой – Sempron. Причем они решили что

конкурировать они должны с процессорами Celeron. Поэтому их процессоры маркируются

рейтингом, обозначающим частоту соответствующего по производительности Intel Celeron.

Например самый простой Sempron в этом модельном ряду имеет рейтинг 2100+ при частоте 1500 МГц, для сравнения – Athlon XP на такой частоте имеет рейтинг 1800+.

Как итог – процессоры Sempron для Socket 462 представляют из себя старый Athlon XP, но с новой рейтинговой маркировкой.

Athlon 64 FX

Тактовые частоты – 2,4...2,8 ГГц.

Частота FSB - 200 МГц.

Частота HTT – 800 (4x200), 1000 (4x250) МГц.

Процессорный разъем - Socket 939.

Кэш-память L1 - 128 Кб.

Кэш-память L2 - 1024 Кб.

Технология изготовления - 0,13 мкм, 0,09 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2, 3DNow!,



Кодовое наименование этих процессоров – K8. Процессоры Athlon 64 FX можно

сравнить

с процессорами Pentium Extreme Edition – они являются предвестниками больших изменений в модельных рядах процессоров. Первые Athlon 64 FX с индексами 53 и 55 представляли из себя адаптированные под новый Socket 939 серверные процессоры Opteron 100-й серии с ядром Clawhammer. Они имели массу новинок, которые выгодно их отличают от предыдущих процессоров.

Во-первых это встроенный в процессор двухканальный контроллер памяти. Получается что северный мост чипсета встроен в процессор – вполне неплохой шаг, что бы избавиться от необходимости согласовывать параметры этого контроллера с производителями чипсетов, которые нередко оказываются хуже чем это требуется процессору. Например VIA, самый массовый производитель чипсетов для AMD, с завидной постоянностью выпускает чипсеты с проблемами контроллера памяти, а потом героически их преодолевает.

Второе что стоит отметить, это шина **HyperTransport**, или **HTT**. Так как потребность в использовании северного моста чипсета исчезла, появилась необходимость каким-то образом поддерживать связь с остальными контроллерами чипсета – контроллером AGP и южным мостом. Для этих целей и используется шина HTT, контроллер которой встроен в процессор.

Она имеет суммарную разрядность 32 бита, и работает в зависимости от модели на частотах

800 (4x200) МГц или 1000 (4x250) МГц. Соответственно имеет пропускную способность 3200

Мб/с или 4000 Мб/с. Этого вполне достаточно для обмена данными с AGP и южным мостом.

Третье – этот процессор поддерживает технологию AMD64, т.е. является 64-разрядным (что видно из названия), и технологию NX-bit, о которой мы уже говорили.

Athlon 64

Тактовые частоты – 1,8...2,4 ГГц.

Частота FSB - 200 МГц.

Частота HTT – 800 (4x200), 1000 (4x250) МГц.

Процессорный разъем - Socket 939, Socket 754.

Кэш-память L1 - 128 Кб.

Кэш-память L2 – 512 Кб, 1024 Кб.

Технология изготовления - 0,13 мкм, 0,09 мкм.

Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2, 3DNow!, SSE3.

Первые выпуски Athlon 64 для Socket 939 представляли из себя тот же 0,13 мкм ClawHammer что и применялся в Athlon 64 FX, но с меньшими частотами, к тому-же модель для Socket 754 имеет всего один контроллер памяти. В модельном ряду этих процессоров есть ещё одно ядро – 0,13 мкм Newcastle, которое отличается от предшественника только уменьшенным кэшем L2 до 512 Кб, что серьёзно удешевило процессор. Причем он выпускается как для Socket 939, так и для



Socket 754, только с разным количеством контроллеров памяти. Последняя версия ядра для этого процессора – Winchester выполнена по технологии 0,09 мкм, имеет 512 Кб кэша L2, поддерживает набор SSE3 и выпускается пока только для Socket 754.

Athlon 64 X2

Тактовые частоты – 2,0...2,4 ГГц.
 Частота FSB - 200 МГц.
 Частота HTT – 1000 (4x250) МГц.
 Процессорный разъем - Socket 939.
 Кэш-память L1 - 128 Кб.
 Кэш-память L2 – 2x1024 Кб, 2x512 Кб.
 Технология изготовления - 0,09 мкм.
 Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2, SSE3, 3DNow!,



Этот процессор – выпускается с применением ядра Manchester - 2x512 Кб L2, или Toledo - 2x1024 Кб. Оба варианта устанавливаются в Socket 939, имеют двойной контроллер памяти, и представляют серьезную конкуренцию Pentium D. Ведь их рейтинги начинаются от 3800+, и достигают 4800+. То есть они сопоставляются процессорам, которых ещё не существует – процессорам с частотой более 4 ГГц.

Sempron

Тактовые частоты – 1,4...2,0 ГГц.
 Частота FSB - 200 МГц.
 Частота HTT – 800 (4x200) МГц.
 Процессорный разъем - Socket 939, Socket 754.
 Кэш-память L1 - 128 Кб.
 Кэш-память L2 - 128 Кб, 256 Кб.
 Технология изготовления - 0,13 мкм, 0,09 мкм.
 Поддерживаемые наборы команд – x86, x87, MMX, SSE, SSE2, 3DNow!, SSE3.



Этот Sempron представляет из себя урезанный вариант ядра Newcastle или Winchester, но только с одним каналом памяти, и с замедленной до 800 МГц шиной HyperTransport. К тому же в некоторых моделях присутствует поддержка AMD64, и практически все поддерживают NX-bit.

3.8. Сводные таблицы характеристик процессоров Intel

3.8.1. Устаревшие модели (не обязательно)

Поколение	Intel		
	Модель	Серия	Характеристики
	8086	x86	5 – 10 МГц Ширины шины 16 бит 10 кратная производительность 8080

	8088		5, 8 MHz Ширина внешней шины 8 бит – данные, 20 – адреса Идентичен 8086
	80186		Использовался в основном во встроенных приложениях — контроллерах, систем точек продажи, терминалах, и тому подобных Включал два таймера, контроллер DMA, и контроллер прерываний на чипе помимо процессора
	80188		Версия 80186 с внешней шиной 8 бит
	80286		6 - 12,5 MHz Ширина шины: 16 бит Добавлена подкачка, аппаратная защита
	DX	80386	16 – 33 MHz Ширина шины – 32 бита Первый чип x86 для поддержки 32-битных наборов команд Переработанная и расширенная поддержка защиты памяти
	SX		16 – 33 MHz Внутренняя архитектура 32 бита Внешняя шина – 16 бит Встроенная мультизадачность
	SL		20 – 25 MHz Внутренняя архитектура 32 бита Внешняя шина – 16 бит Первый чип, изготовленный для портативных компьютеров Включает контроллеры кэша, шины и памяти
	EX		Вариант 80386SX предназначенного для встраиваемых систем Статическое ядро, периферийные устройства, интегрированные в микросхему
	DX	80486	25 – 50 MHz Ширина шины – 32 бита кэш первого уровня на чипе Встроенный математический сопроцессор
	SX		16 – 33 MHz Ширина шины – 32 бита Идентичен по дизайну 486DX но без математического сопроцессора
	DX2		50 – 66 MHz Ширина шины 32 бита Использовал технологию «удвоения скорости» при помощи работы ядра процессора на частоте, удвоенной по сравнению с частотой шины
	SL		20 – 33 MHz Ширина шины – 32 бита Использовался в ноутбуках
	DX4		75 – 100 MHz Ширина шины – 32 бита Использовался в высокопроизводительных десктопах начального уровня и дешёвых ноутбуках
	Pentium	Pentium I	Ядро P5 60 – 66 MHz Ядро P54 75 – 120 MHz Ядро P54C 120 – 200 MHz Ширина шины – 64 бит Ширина шины адреса 32 бита Кэш L1 – 16 КБ Суперскалярная архитектура позволила повысить в 5 раз производительность
	Pentium MMX		166 – 300 MHz Частота системной шины – 66 МГц Кэш L1 – 32 КБ
	Pentium Pro	P6/Pentium M	166 – 200 МГц Частота системной шины – 60, 66 МГц Кэш L1 16 КБ

		Кэш L2 512 КБ или 256 КБ (встроенный) Использовался преимущественно в серверных системах
	Pentium II	233 – 300 МГц для 0,35 мкм 233 – 450 для 0,25 мкм Частота системной шины – 66 МГц Кэш L1 32 КБ Кэш L2 512 КБ
	Celeron (Pentium II)	Covington 266 – 300 МГц Кэш L1 32 КБ Нет кэша L2 Частота системной шины - 66 МГц Mendochino 300 – 500 МГц Кэш L1 32 КБ Кэш L2 128 КБ Частота системной шины - 66 МГц
	Pentium III	Katmai 450 – 600 МГц Кэш L2 512 КБ Частота системной шины - 100 МГц Улучшенный PII, то есть ядро, основанное на P6 с SSE Coppermine 500 – 1000 МГц Кэш L2 256 КБ Частота системной шины - 100 МГц, 133 МГц Tualatin 1133 – 1400 МГц Кэш L1 32 КБ Кэш L2 512 КБ (встроенный) или 256 КБ Частота системной шины - 133 МГц
	Pentium Xeon II, III	PII Xeon 400 - 450 МГц PIII Xeon 500 – 1000 МГц Кэш L2: 256 КБ, 1 МБ или 2 МБ (интегрированный) Ширина системной шины: 64 бит Частота системной шины: 133 МГц (с кэшем L2 256 КБ) или 100 МГц (с кэшем L2 1-2 МБ) Использовался в двухпроцессорных серверах и рабочих станциях (256 КБ L2) или 4- и 8-процессорных серверах (1-2 МБ L2)
	Celeron (Pentium III)	Commermine 533 – 850 МГц Частота системной шины – 66, 100 МГц Кэш L1 32 КБ Кэш L2 128 КБ SSE, Socket 370 Tualatin 1.0 – 1.4 ГГц Частота системной шины – 100 МГц Кэш L1 32 КБ Кэш L2 256 КБ
	Pentium M	Banias 900 – 1700 МГц Частота системной шины: 400 МГц (Netburst) Кэш L1: 64 КБ Кэш L2: 1 МБ (встроенный) Базируется на ядре Pentium III

			Dothan 1,0 – 2,26 ГГц Кэш L2: 2 МБ Исправленный модуль предвыборки данных
	Celeron M		Banias 310 – 1500 МГц Кэш L1: 64 КБ Кэш L2: 512 КБ (интегрированный) SSE2 SIMD-инструкции Нет поддержки технологии SpeedStep
	Intel Core		Dothan 350 – 1700 МГц Кэш L1: 64 КБ Кэш L2: 1 МБ (интегрированный) SSE2 SIMD-инструкции Нет поддержки технологии SpeedStep
	Dual-Core Xeon LV		1.5 – 2.33 ГГц Частота системной шины: 667 МГц Удвоенное (или одиночное в случае Solo) ядро с разделяемым кэшем L2 размером 2 МБ SSE3 SIMD-инструкции
		32-битные NetBurst	2.0 ГГц Частота системной шины: 667 МГц Разделяемый кэш L2 размером 2 МБ Основан на ядре Yonah, с поддержкой SSE3 SIMD-инструкций
	Pentium 4		1.4 – 3.4 ГГц Частота системной шины: 400 МГц или Частота системной шины: 533, 800 МГц (все версии включают в себя Hyper Threading) L2-кэш — интегрированный 256, 512 КБ SSE2 SIMD Extensions
	Xeon		1.4, 1.5, 1.7 от 2.0 до 3.6 ГГц Кэш L2: 256 КБ Advanced Transfer Cache (интегрированный) Частота системной шины: 400 МГц Поддержка расширений SIMD: SSE2
	Mobile Pentium 4-M		1,4 – 2,6 ГГц Частота системной шины: 400 МГц Интегрированный L2-кэш 512 КБ Поддерживает до 1 ГБ DDR 266 МГц оперативной памяти Поддерживает системы управления энергосбережением ACPI 2.0 и APM 1.2
	Pentium 4EE		Построен на основе ядра «Gallatin» Xeon, но с 2Мб кэшем
	Pentium 4E		L2 кэш: 1 МБ Частота системной шины: 533 МГц (только 2.4А и 2.8А) Частота системной шины: 800 МГц (все остальные модели) Поддержка Hyper-Threading реализована только в процессорах с 800МГц системной шиной Процессоры серии 6xx имеют кэш L2 размером 2 МБ и поддержку EM64T
	Pentium 4F		3,2—3,6 ГГц Ядро то же, что и у модели 4E «Prescott» Начиная со степпинга D0, этот процессор также поддерживает EM64T 64-битные расширения
	Itanium	64-битные процессоры: IA-64	733 МГц и 800 МГц
	Itanium 2		900 МГц и 1 ГГц

3.8.2. Современные модели

Поколение	Intel		
	Модель	Серия	Характеристики
	Pentium D	64-битные процессоры: EM64T — NetBurst	<p>Двухъядерный (Dual-core) микропроцессор Частота системной шины: 800 (4x200) МГц Частота системной шины: 800 (4x200) МГц</p> <p>Smithfield 2,8—3,4 ГГц Кэш L2: 1 МБ x 2 Производительность увеличилась примерно на 60 % по сравнению с одноядерным микропроцессором Prescott</p> <p>Presler 2,8—3,6 ГГц Кэш L2: 2 МБ x 2</p>
	Pentium Extreme Edition		<p>Двухъядерный (Dual-core) микропроцессор Частота системной шины: 1066 (4x266) МГц Поддержка Hyper-Threading</p> <p>Smithfield Pentium 840 EE, 3,20 ГГц кэш L2 размером 2 x 1 МБ</p> <p>Presler Pentium 955 EE, 3,46 ГГц Pentium 965 EE, 3,73 ГГц L2 кэш: 2 МБ x 2</p>
	Xeon		<p>Nocona</p> <p>Irwindale</p> <p>Cranford MP версия микропроцессора</p> <p>Potomac Отличается от микропроцессора Cranford только наличием кэша L3 размером 8 МБ</p> <p>Paxville DP 2,8 ГГц Двухъядерная версия микропроцессора Irwindale, имеющая кэш L2 размером 4 МБ</p> <p>Paxville MP 2.67 — 3.0 ГГц Серия Dual-Core Xeon 7000 Версия микропроцессора</p> <p>Paxville DP с поддержкой MP Кэш L2: 2 МБ (по 1 МБ на ядро) или 4 МБ (по 2 МБ на ядро)</p> <p>Dempsey 2.67 — 3.73 ГГц Серия Dual-Core Xeon 5000 MP версия микропроцессора Presler Кэш L2: 4 МБ (по 2 МБ на ядро) Упаковка процессора: Socket J</p>

	Xeon	64-битные процессоры: EM64T — Intel Core	<p>Woodcrest От Xeon 5110 — 1,60 ГГц (4 Мб L2, 1066 МГц FSB, 65 Вт) до Xeon 5160 — 3,00 ГГц (4 Мб L2, 1333 МГц FSB, 80 Вт) Микропроцессор для серверов и рабочих станций с поддержкой симметричной многопроцессорности (SMP) (в случае двухпроцессорных систем) Двухъядерный Dual-Core микропроцессор Поддержка инструкций SIMD: SSE4 Intel Virtualization Technology EIST (Enhanced Intel SpeedStep Technology) iAMT2 (Intel Active Management Technology) — удаленное управление компьютерами</p> <p>Clovertown Xeon L5320 — 1,86 ГГц (2x4 Мб L2, 1066 MHz FSB, 40 Вт) - Xeon X5355 — 2,66 ГГц (2x4 Мб L2, 1333 MHz FSB, 105 Вт) Микропроцессор для серверов и рабочих станций с поддержкой симметричной многопроцессорности (SMP) (в случае двухпроцессорных систем) Четырехъядерный Quad-Core микропроцессор Intel Virtualization Technology EIST (Enhanced Intel SpeedStep Technology) SSSE3 SIMD instructions iAMT2 (Intel Active Management Technology) — удаленное управление компьютерами</p>
	Intel Core 2		<p>Conroe Core 2 Duo E4200 — 1,60 ГГц (2 Мб L2, 800 МГц FSB, нет VT) - Core 2 Duo E4200 — 1,60 ГГц (2 Мб L2, 800 МГц FSB, нет VT) Микропроцессор для настольных систем с поддержкой симметричной многопроцессорности (SMP), ограниченной двумя микропроцессорам Поддержка инструкций SIMD: SSE3 Intel Virtualization Technology LaGrande Technology EIST (Enhanced Intel Speed Step Technology) iAMT2 (Intel Active Management Technology) — удаленное управление компьютерами Сокет: LGA775</p> <p>Conroe XE Core 2 Extreme X6800 — 2,93 ГГц (4 Мб L2, 1066 МГц FSB) - Core 2 Extreme X6900 — 3,20 ГГц (4 Мб L2, 1066 МГц FSB) Микропроцессор для настольных систем серии eXtreme Edition с поддержкой симметричной многопроцессорности (SMP) (ограничено двумя микропроцессорами) Реализованы те же технологии, что и у микропроцессора Conroe Сокет: LGA775</p> <p>Merom Core 2 Duo U7500 — 1,06 ГГц (2 Мб L2, 533 МГц FSB) - Core 2 Duo T7700 — 2,40 ГГц (4 Мб L2, 800 МГц FSB) Мобильный микропроцессор с поддержкой симметричной многопроцессорности (SMP) (ограничено двумя микропроцессорами) Реализованы те же технологии, что и у микропроцессора Conroe Сокет: Socket 479</p>
	Pentium Dual Core		<p>Intel Pentium E2220 - 2.40 ГГц (1 Мб L2, 800 МГц FSB) - Intel Pentium E2140 - 1.60 ГГц (1 Мб L2, 800 МГц FSB) Микропроцессор для настольных систем с поддержкой симметричной многопроцессорности (SMP), ограниченной двумя микропроцессорам Поддержка инструкций SIMD: SSE3 LaGrande Technology Execute Disable Bit EIST (Enhanced Intel Speed Step Technology) iAMT2 (Intel Active Management Technology) — удаленное управление</p>

		компьютерами LGA775
	Intel Atom	Intel Atom Z540 - 1.86 ГГц (512 Кб L2, 533 МГц FSB, 2.4 Вт TDP) - Intel Atom Z500 - 800 МГц (512 Кб L2, 400 МГц FSB, 0.65 Вт TDP) Микропроцессор для ультрамобильных систем с поддержкой симметричной многопроцессорности (SMP), ограниченной двумя микропроцессорам Поддержка инструкций SIMD: SSE3 EIST (Enhanced Intel Speed Step Technology) Deep Power Down Technology (C6) Hyper-Threading (начиная с моделей с 533 MHz FSB) FCBGA8

3.9. Сводные таблицы характеристик процессоров AMD

3.9.1. Устаревшие модели (не обязательно)

Поколение	AMD		
	Модель	Серия	Характеристики
	Am8088	x86	Аналог процессора Intel 8088.
	Am80C88		Аналог процессора Intel 80C88 (выпускался по технологии CMOS).
	Am8086		Аналог процессора Intel 8086.
	Am80C86		Аналог процессора Intel 80C86 (выпускался по технологии CMOS).
	Am80188		Аналог процессора Intel 80188.
	Am80L188		Am80188 для встраиваемых систем.
	Am80186		Аналог процессора Intel 80186.
	Am80L186		Am80186 для встраиваемых систем.
	Am186EM		Модернизированный Am80186 для встраиваемых систем.
	Am80286		Аналог процессора Intel 80286. тактовая частота: 12-16 МГц
	Am80C286		Аналог процессора Intel 80C286 (выпускался по технологии CMOS).
	Am80EC286		Am80C286 с пониженным энергопотреблением.
	Am80L286		Am80286 для встраиваемых систем.
	Am386DX	Am386	Базовый процессор семейства. Функциональный аналог процессора Intel 80386DX. Практически полный аналог Intel-овской "тройки". Кодовое имя: P9. Тех. характеристики: 275000 транзисторов; тактовая частота: 16-32 МГц; процессор 32-разрядный; шина данных 32-разрядная (16-32МГц); адресная шина 32-разрядная; общая разрядность: 32.
	Am386DXL		Am386DX с пониженным тепловыделением.
	Am386DXLV		Am386DX с пониженным напряжением питания.
	Am386SX		Am386 с 16-разрядной внешней шиной данных. Low-End версия AMD Am 386™ DX. Кодовое имя: P9. Тех. характеристики: 275000 транзисторов; тактовая частота: 16-32 МГц; процессор 32-разрядный; шина данных 16-разрядная (16-32МГц); адресная шина 24-разрядная; общая разрядность: 16
	Am386SXL		Am386SX с пониженным тепловыделением.
	Am386SXLV		Am386SX с пониженным напряжением питания.
	Am386DE		Am386DX для встраиваемых систем.
	Am386SE		Am386SX для встраиваемых систем.
	Am386EM		Модернизированный Am386DX для встраиваемых систем с

			интегрированным контроллером памяти.
	Am486DX	Am486	Базовый процессор семейства. Функциональный аналог процессора Intel 80486. Процессор со встроенными кэшем первого уровня и математическим сопроцессором (FPU). Немного отставал по производительности от аналогичного процессора фирмы Intel. Кодовое имя: P4 :) Тех. характеристики: 1,25 млн. транзисторов; тактовая частота: 25-50 МГц; кэш первого уровня: 8 Кб; кэш второго уровня на материнской плате (до 512 Кб); процессор 32-разрядный; шина данных 32-разрядная (20-50МГц); адресная шина 32-разрядная; общая разрядность: 32
	Am486DXL		Am486DX с пониженным тепловыделением.
	Am486DXLV		Am486DX с пониженным напряжением питания.
	Am486DX2		Am486DX с внутренним удвоением частоты. Полностью 32-х разрядный процессор. Кодовое имя: P24. Тех характеристики: 1,25 млн. транзисторов; тактовая частота: 50-66 МГц; кэш первого уровня: 8 Кб; кэш второго уровня на материнской плате (до 512 Кб); процессор 32-разрядный; шина данных 32-разрядная (25-33 МГц); адресная шина 32-разрядная; общая разрядность: 32
	Am486DX2WB		Am486DX2 с кэш-памятью с обратной записью.
	Am486DXL2		Am486DX2 с пониженным тепловыделением.
	Enhanced Am486DX2		Am486DX2WB, произведённый по обновлённому техпроцессу.
	Am486DX4		Am486DX с внутренним утроением частоты. Последняя "четвёрка" от AMD с повышенной тактовой частотой. Кодовое имя: P24C. Тех характеристики: 1,25 млн. транзисторов; тактовая частота: 75-120 МГц; кэш первого уровня: 8 Кб; кэш второго уровня на материнской плате (до 512 Кб); процессор 32-разрядный; шина данных 32-разрядная (25-40 МГц); адресная шина 32-разрядная; общая разрядность: 32.
	Am486DX4WB		Am486DX4 с кэш-памятью с обратной записью.
	Enhanced Am486DX4		Am486DX4WB, произведённый по обновлённому техпроцессу.
	Am486DX4SE		Am486DX4 для встраиваемых систем.
	Am486SX		Am486DX без встроенного математического сопроцессора.
	Am486SXLV		Am486SX с пониженным напряжением питания.
	Am486SE		Am486SX для встраиваемых систем.
	Am486SX2		Am486SX с внутренним удвоением частоты.
	Am5x86		Модернизированный Am486DX с увеличенным до 16 Кб кэшем первого уровня и внутренним умножением частоты на 4.
	Élan		Семейство встраиваемых микропроцессоров.
	5k86	K5	Первый процессор серии K5. Первый процессор x86 компании AMD, имеющий внутреннюю архитектуру CISC-to-RISC. SSA/5 Этот процессор показывал отличную производительность в офисных приложениях, но имел слабый FPU, впрочем как и предыдущий. Для маркировки этих процессоров тоже использовался PR-рейтинг. Кодовое имя: 5k86. Тех. характеристики: 4,3 млн. транзисторов; технология производства: 0,35 мкм; тактовая частота: 90-133 МГц; кэш первого уровня: 24 Кб (8 Кб на данные и 16 Кб на инструкции); кэш второго уровня на материнской плате (до 1 Мб); процессор 64-разрядный; шина данных 64-разрядная (60-66 МГц); адресная шина 32-разрядная; общая разрядность: 32; разъем Socket 7.
	K5		Модернизированный 5k86.

			Godot
	K6	K6	<p>K6 Первый процессор серии K6. До приобретения AMD компании NexGen разрабатывался как NexGen Nx686. Процессор, построенный по x86-to-RISC86 технологии, может выполнять до 6 инструкций RISC86 одновременно. Он устанавливается в разъем Socket 7 и может быть использован в платах, предназначенных для процессоров Pentium. В отличие от своих собратьев - процессоров Pentium MMX и Cyrix 6x86MX, он программно совместим с процессором Pentium Pro и работает с MMX инструкциями, что делает его сравнимым с процессором Pentium II фирмы Intel. Был создан на базе дизайна процессора 686 от приобретенной AMD компании NexGen. Кодовое имя: K6. Тех. характеристики: 888 млн. транзисторов; технология производства: 0835 мкм; тактовая частота: 166-233 МГц; кэш первого уровня: 64 Кб (32 Кб на данные и 32 Кб на инструкции); кэш второго уровня на материнской плате (до 1 Мб); процессор 64-разрядный; шина данных 64-разрядная (66 МГц); адресная шина 32-разрядная; общая разрядность: 32; разъем Socket 7</p> <p>Little Foot K6, произведённый по обновлённому техпроцессу. Этот процессор выпускался по 0.25 мкм технологическому процессу и имел более высокую тактовую частоту, чем предшественник. Кодовое имя: Little Foot. Тех. характеристики: 8.8 млн. транзисторов; технология производства: 0,25 мкм; тактовая частота: 233-300 МГц; кэш первого уровня: 64 Кб (32 Кб на данные и 32 Кб на инструкции); кэш второго уровня на материнской плате (до 1 Мб); процессор 64-разрядный; шина данных 64-разрядная (66 МГц); адресная шина 32-разрядная; общая разрядность: 32; разъем Socket 7.</p>
	K6-2		<p>В этом процессоре основными усовершенствованиями являются поддержка дополнительного набора инструкций 3DNow!, который существенно повышает производительность в оптимизированных программах и играх, а также 100-МГц системная шина. Кодовое имя: Chomper XT. Тех. характеристики: 9.3 млн. транзисторов; технология производства: 0.25 мкм; тактовая частота: 266-550 МГц; кэш первого уровня: 64 Кб (32 Кб на данные и 32 Кб на инструкции); кэш второго уровня на материнской плате (до 1 Мб); процессор 64-разрядный; шина данных 64-разрядная (66-100 МГц); адресная шина 32-разрядная; общая разрядность: 32; разъем Socket 7.</p> <p>Chomper Модернизированное ядро Little Foot с блоком 3DNow!</p> <p>CXT Chomper Extended — ядро Chomper с более высокой тактовой частотой.</p>
	K6-III		<p>Первый процессор от AMD, имеющий кэш-память второго уровня, объединенную с ядром. Представляют собой K6-2 с 256 Кбайт кэш-памятью L2 на чипе, работающей на той же частоте, что и ядро процессора. Рекомендуется для установки на материнские платы Super Socket 7, имеющие поддержку AGP. Кодовое имя: Sharptooth. Тех. характеристики: 21.3 млн. транзисторов; технология производства: 0.25 мкм; тактовая частота: 350-500 МГц; кэш первого уровня: 64 Кб (32 Кб на данные и 32 Кб на инструкции); кэш второго уровня 256 Кб (полноскоростной);</p>

			<p>кэш третьего уровня на материнской плате (до 3 Мб); процессор 64-разрядный; шина данных 64-разрядная (100 МГц); адресная шина 32-разрядная; общая разрядность: 32; разъем Super Socket 7.</p> <p>Sharptooth</p> <p>Модернизированное ядро Little Foot с интегрированным кэшем второго уровня (256 Кб).</p>
	K6-III+		<p>Sharptooth</p> <p>Мобильный вариант, произведённый по обновлённому техпроцессу, поддерживающий технологию PowerNow! и имеющий расширенный набор инструкций 3DNow!</p>
	K6-2+		<p>Sharptooth</p> <p>K6-III+ с уменьшенным кэшем второго уровня (128Кб).</p> <p>Последний процессор из семейства K6® выполнен по 0,18 мкм технологическому процессу, имеет кэш-память второго уровня размером 128 Кбайт и технологию PowerNow!™. Тех. характеристики: технология производства: 0.18 мкм; тактовая частота: 450-550 МГц; кэш первого уровня: 64 Кб (32 Кб на данные и 32 Кб на инструкции); кэш второго уровня на материнской плате (до 3 Мб); процессор 64-разрядный; шина данных 64-разрядная (95-100 МГц); адресная шина 32-разрядная; общая разрядность: 32; разъем Super Socket 7.</p>
		K7	<p>Первый процессор, архитектура и интерфейс которого отличаются от Intel. После его выхода позиции Intel несколько пошатнулись, т. к. он демонстрировал большую производительность в большинстве приложений, чем Pentium III при равных тактовых частотах. Имеет расширенный набор инструкций Enhanced 3DNow!. Кодовое имя: K7, K75 (алюминиевые соединения), K76 (медные соединения). Тех. характеристики: 22 млн. транзисторов; технология производства: 0.25-0.18 мкм; тактовая частота: 500-1000 МГц; кэш первого уровня: 128 Кб (64 Кб на данные и 64 Кб на инструкции); кэш второго уровня 512 Кб, работающий на 1/2, 2/5 или 1/3 частоты процессора; процессорная шина – Alpha EV-6 200 МГц (DDR 100x2); общая разрядность: 32; разъем Slot A.</p>
	Athlon		<p>Argon (K7)</p> <p>Первое ядро, использованное в процессорах Athlon. Имеет внешний инклюзивный кэш второго уровня (512 Кб).</p> <p>Orion/Pluto (K75)</p> <p>Ядро Argon, выполненное по обновлённому техпроцессу.</p> <p>Thunderbird</p> <p>Ядро K75 с интегрированным эксклюзивным кэшем второго уровня (256 Кб).</p> <p>Этот процессор выпущен по технологии 0,18 мкм с использованием технологии медных соединений. Первоначально выпускался в форм-факторе Slot A, позднее Socket A. На чипе интегрированы 256 Кбайт кэша второго уровня, работающего на частоте процессора. Кодовое имя: Thunderbird. Тех. характеристики: технология производства: 0.18 мкм; тактовая частота: 600-1400 МГц; кэш первого уровня: 128 Кб (64 Кб на данные и 64 Кб на инструкции); кэш второго уровня 256 Кб (полноскоростной); процессорная шина – Alpha EV-6 200-266МГц (DDR 100x2-133x2); общая разрядность: 32; разъем Slot A, позднее Socket A.</p>
	Athlon XP		<p>Версия процессора на ядре Palomino для настольных компьютеров. При маркировке этих процессоров используется не реальная тактовая частота, а индекс</p>

		<p>производительности, т. е. показывается какому Pentium 4 соответствует данный процессор. Например Athlon XP 2000+ работает на частоте 1667 МГц. В отличие от AMD K5, это реальный показатель и Athlon XP 1900+ действительно не уступает P4 1900 МГц, а в некоторых приложениях даже превосходит его. Тех. характеристики: технология производства: 0.18 мкм; тактовая частота: 1333-1800МГц; кэш первого уровня: 128 Кб (64 Кб на данные и 64 Кб на инструкции); кэш второго уровня 256 Кб (полноскоростной); процессорная шина – Alpha EV-6 266МГц (DDR 133х2); общая разрядность: 32; разъем Socket A.</p> <p>Palomino Модернизированное ядро Thunderbird с аппаратной предвыборкой данных и блоком SSE.</p> <p>Thoroughbred Ядро Palomino, выполненное по обновлённому техпроцессу. Продолжение развития процессора Athlon XP. В отличие от предыдущего выполнен по 0,13 мкм тех. процессу и маркировка нанесена не на кристалл, а на специальную пластину. Ядро процессора стало несколько прочнее. При маркировке этих процессоров также используется не реальная тактовая частота, а индекс производительности. Кодовое имя: Thoroughbred. Тех. характеристики: технология производства: 0.13 мкм; тактовая частота: 1733-2133МГц; кэш первого уровня: 128 Кб (64 Кб на данные и 64 Кб на инструкции); кэш второго уровня 256 Кб (полноскоростной); процессорная шина – Alpha EV-6 266МГц (DDR 133х2); общая разрядность: 32; разъем Socket A.</p> <p>Barton Модернизированное ядро Thoroughbred с увеличенным до 512 Кб кэшем второго уровня.</p> <p>Thorton Ядро Barton с частично отключённым кэшем второго уровня (256 Кб).</p>
	Athlon MP	<p>Palomino Thoroughbred Thorton</p> <p>Процессор Athlon XP с возможностью работы в многопроцессорной конфигурации.</p>
	Athlon 4	<p>Corvette Мобильный вариант ядра Palomino с поддержкой энергосберегающей технологии PowerNow!</p>
	Mobile Athlon XP	<p>Thoroughbred Мобильный вариант ядра Thoroughbred с поддержкой энергосберегающей технологии PowerNow!</p>
	Duron	<p>Spitfire Ядро Thunderbird с меньшим кэшем второго уровня (64 Кб). Low-End версия Athlon™ Thunderbird с урезанным до 64 Кбайт кэшем второго уровня. Разносит Celeron в "пух и прах", хотя обладает меньшей ценой. Кодовое имя: Spitfire. Тех. характеристики: 25 млн. транзисторов; технология производства: 0.18 мкм; тактовая частота: 600-950 МГц; кэш первого уровня: 128 Кб (64 Кб на данные и 64 Кб на инструкции); кэш второго уровня 64 Кб (полноскоростной); процессорная шина – Alpha EV-6 200МГц (DDR 100х2); общая разрядность: 32; разъем Socket A.</p>

			<p>Morgan Ядро Palomino с меньшим кэшем второго уровня (64 Кб). Этот Duron выполнен на ядре Morgan - урезанном варианте Palomino (кэш L2 не 256, а 64 Кбайта). Кодовое имя: Morgan. Тех. характеристики: 25.18 млн. транзисторов; технология производства: 0.18 мкм; тактовая частота: 1000-1300 МГц; кэш первого уровня: 128 Кб (64 Кб на данные и 64 Кб на инструкции); кэш второго уровня 64 Кб (полноскоростной); процессорная шина – Alpha EV-6 200МГц (DDR 100х2); общая разрядность: 32; разъем Socket A.</p> <p>Applebred Ядро Thoroughbred с частично отключённым кэшем второго уровня (64 Кб).</p>
	Mobile Duron		<p>Мобильная версия Duron-а с технологией PowerNow!™. Тех. характеристики: технология производства: 0.18 мкм; тактовая частота: 700-950 МГц; кэш первого уровня: 128 Кб (64 Кб на данные и 64 Кб на инструкции); кэш второго уровня 64 Кб (полноскоростной); процессорная шина – Alpha EV-6 200МГц (DDR 100х2); общая разрядность: 32.</p> <p>Camaro Мобильный вариант ядра Spitfire с поддержкой энергосберегающей технологии PowerNow!</p> <p>Morgan Мобильный вариант ядра Morgan с поддержкой энергосберегающей технологии PowerNow!</p>
	Sempron		<p>Thoroughbred Thorton Barton</p> <p>Переименованный Athlon XP, предназначенный для рынка недорогих компьютеров.</p>
	Geode NX		<p>Thoroughbred Процессор для встраиваемых систем.</p>
	Geode GX	Geode	Переименованный NS Geode GX2. Имеет встроенные контроллеры памяти и видео.
	Geode LX		Модернизированный Geode GX.

3.9.2. Современные модели

Поколение	AMD		
	Модель	Серия	Характеристики
	Opteron	K8 Все процессоры серии K8 имеют интегрированный контроллер памяти (одноканальный DDR — Socket 754, двухканальный DDR — Socket 939 / Socket 940 или двухканальный DDR2 — Socket AM2 / Socket F) и поддерживают набор инструкций AMD64 (если не указано обратное).	Sledgehammer Первая модель процессоров Opteron (130 нм техпроцесс). Venus Одноядерные процессоры Opteron 1xx (90 нм техпроцесс). Troy Одноядерные процессоры Opteron 2xx (90 нм техпроцесс). Athens Одноядерные процессоры Opteron 8xx (90 нм техпроцесс). Denmark Двухядерные процессоры Opteron 1xx (90 нм техпроцесс). Italy Двухядерные процессоры Opteron 2xx (90 нм техпроцесс). Egypt Двухядерные процессоры Opteron 8xx (90 нм техпроцесс). Santa Ana Двухядерные процессоры Opteron (90 нм техпроцесс, Socket AM2). Santa Rosa Двухядерные процессоры Opteron (90 нм техпроцесс, Socket F).
	Athlon 64		Clawhammer Первая модель процессоров Athlon 64 (130 нм техпроцесс, 1 Мб кэша второго уровня). Newcastle Ядро Clawhammer с частично отключённым кэшем второго уровня (512 Кб). Winchester Процессоры Athlon 64, произведённые по обновлённому техпроцессу (90 нм). Venice Ревизия ядра Winchester. San Diego Ревизия ядра Venice. Orleans Процессоры Athlon 64 для Socket AM2. Lima Одноядерные процессоры на базе ядра Brisbane.
	Athlon 64 FX		Sledgehammer Первая модель процессоров Athlon 64 FX (130 нм техпроцесс). San Diego Процессоры Athlon 64 FX, произведённые по обновлённому техпроцессу (90 нм). Toledo Двухядерные процессоры Athlon FX (90 нм).
	Athlon 64 X2		Manchester Двухядерные процессоры на базе ядра Venice (512 Кб кэша второго уровня, Socket 939). Toledo Двухядерные процессоры на базе ядра Venice (1 Мб кэша второго уровня, Socket 939). Windsor Двухядерные процессоры на базе ядра Orleans (1 Мб кэша второго уровня, Socket AM2). Brisbane Двухядерные процессоры, произведённые по обновлённому техпроцессу (65 нм).
	Athlon X2		Brisbane

Переименованные процессоры Athlon 64 X2 с новой

			системой обозначения моделей.
	Sempron		<p>Paris Первая модель процессоров Sempron K8. Ядро Newcastle с частично отключённым кэшем второго уровня (256 Кб). Инструкции AMD64 заблокированы.</p> <p>Palermo Ядро Winchester с частично отключённым кэшем второго уровня (128 или 256 Кб).</p> <p>Manila Ядро Orleans с частично отключённым кэшем второго уровня (256 Кб).</p> <p>Sparta Ядро Lima с частично отключённым кэшем второго уровня (256 Кб).</p>
	Athlon XP-M		Dublin Мобильные процессоры. Инструкции AMD64 заблокированы.
	Mobile Athlon 64		<p>Newcastle Мобильный вариант ядра Newcastle.</p> <p>Odessa Процессоры Mobile Athlon 64, произведённые по обновлённому техпроцессу (90 нм).</p> <p>Oakville Процессоры Mobile Athlon 64 LV (их наследниками стали Turion 64), произведённые по обновлённому техпроцессу (90 нм) с пониженным энергопотреблением.</p> <p>Newark Процессоры Mobile Athlon 64, пришли на смену Odessa с Socket 754 и поддержкой SSE3.</p> <p>Trinidad Двухядерные процессоры Mobile Athlon 64 X2 (90 нм техпроцесс, арх. K8 rev.F, 512 Кб кэша второго уровня).</p>
	Turion 64		<p>Lancaster Первая модель процессоров Turion 64 (90 нм техпроцесс).</p> <p>Sherman Процессоры Turion 64, произведённые по обновлённому техпроцессу (65 нм).</p>
	Turion 64 X2		<p>Taylor Двухядерные процессоры Turion 64 X2 (90 нм техпроцесс, 256 Кб кэша второго уровня). Socket S1.</p> <p>Tyler Процессоры Turion 64 X2, произведённые по обновлённому техпроцессу (65 нм). Socket S1.</p>
	Mobile Sempron		<p>Georgetown Первая модель процессоров Mobile Sempron (90 нм техпроцесс, Socket 754).</p> <p>Albany Пришел на смену Georgetown, отличается поддержкой SSE3</p> <p>Richmond Пришел на смену Albany, отличается двухканальным контроллером памяти DDR2 и разъемом Socket AM2 (арх. K8 rev.F)</p>
	Phenom	<p>K10 Процессоры серии K10 имеют интегрированный контроллер памяти (двухканальный DDR2, имеется поддержка</p>	<p>Agena (B2) Первая модель настольных процессоров Phenom серии 9x00 (65 нм техпроцесс). Содержит досадную ошибку TBL bug</p> <p>Agena (B3) Новая ревизия настольный процессоров Phenom серии 9x50. характеризуется повышенными частотами и исправленной TBL bug</p> <p>Toliman Трёхядерные процессоры серии Phenom 8x00 (65 нм</p>

		DDR3), разделяемый кэш третьего уровня и поддерживают набор инструкций AMD64	техпроцесс). Deneb Четырехядерные процессоры Phenom (45 нм техпроцесс). Prorus Четырехядерные процессоры Phenom (45 нм техпроцесс). Отличаются отсутствием кэше L3 Нека Трехядерные процессоры серии Phenom (45 нм техпроцесс). Regor Двухядерные процессоры Phenom (45 нм техпроцесс).
--	--	--	--