МИНИСТЕРСТВО ОБЩЕГО И ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ РФ

ОРЕНБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Кафедра ПОВТАС

**Реферат**

Проблемы повышения производительности вычислительных систем и методы их преодоления

|  |  |
| --- | --- |
| Выполнил: | Чукин В.В.группа 98повт-3 |
| Проверил: | Воронков А.Е. |

Оренбург 1999

**Содержание.**

|  |  |
| --- | --- |
| **1.Немного истории.** | **3** |
| **2.Повышение производительности в компьютерных сетях.** | **6** |
| **3.Повышение производительности процесоров** | **8** |
| **4. ММХ-революция в микропроцессорной эволюции.** | **9** |
| **5.Основные проблемы.** | **12** |
| **6.Одно из перспективных решений.** | **12** |
| **7.Влияние архитектуры микропроцессоров на повышение производительности.** | **14** |
| **8.Список литературы.** | **26** |

**Немного истории.**

Электронный компьютер был разработан в 50-е годы как устройство для производства быстрых вычислений. В то время никто не задумывался над его информационными приложениями и усилия разработчиков были направлены в основном на повышение быстродействия компьютера.

В 60-е годы была сделана попытка использовать компьютер для осуществления информационных операций индексирования и анализа текста. С помощью соответствующего программного обеспечения в память компьютера стали записывать тексты и искать возможности оперирования с ними. В настоящее время существует множество фирм, производящих компьютеры. Это, например, Dell, Acer, IBM. Для тех, кто профессионально с компьютером не связан, расскажем очень кратко о некоторых принципах функционирования компьютера.

Структурно компьютер состоит из пяти основных функциональных блоков, объединенных общей задачей быстро производить арифметические и логические действия над числами. Характер действий и их последовательность определяются программой, представляющей собой определенным образом организованную совокупность машинных операций, называемых командами. Руководит работой компьютера устройство управления. Оно получает команды из памяти, декодирует их и генерирует необходимые для их выполнения сигналы. Каждая команда в памяти находится по определенному адресу, который указывается программным счетчиком, находящимся в устройстве памяти. Для запоминания команд в устройстве управления имеется специальный регистр команд.

Устройств ввода и вывода служат для преобразования информации с тех языков программирования и тех скоростей, на которых работает компьютер, на те, которые воспринимает человек или другая система, работающая с данным компьютером. Важнейшим функциональным блоком компьютера является процессор. В нашей стране получили широкое распространение процессоры фирм Intel, AMD и др.

Команды от оператора поступают через устройство ввода в регистр инструкций. По этим инструкциям блок управления вырабатывает команды для всех функциональных блоков компьютера Система регистров общего пользования служит оперативной памятью для поступающих на вход данных.

Арифметическое устройство производит арифметические и логические операции над поступающими от регистра общего назначения данными. О том, какие именно следует выполнять функции, говорят команды блока управления.

Адресный регистр обеспечивает правильную адресацию данных в системе памяти, он также снабжает каждую следующую инструкцию указаниями адресов памяти, где она будет запоминаться.

Одним из ключевых функциональных блоков компьютера является программный счетчик, назначение которого состоит в обновлении каждой инструкции и временном запоминании ее номера. Если записанная программа перейдет в другой инструктивный номер, это изменение будет зафиксировано программным счетчиком. Информация об этом изменении поступает также в регистр инструкций. При каждом следующем шаге рабочей программы происходит обращение к программному счетчику с запросом о выполнении соответствующей операции.

Каждая ячейка памяти компьютера запоминает одно машинное слово. Возможности компьютера определяются числом команд, которое можно ввести в оперативную память. Максимальная емкость оперативной памяти в миникомпьютерах не превышает обычно 32M слов, в суперкомпьютерах емкость оперативной памяти достигает 1000M слов (M = 1 048 576).

Резкое повышение производительности миникомпьютеров при сравнительно небольшом удорожании привело к созданию мегамини- или супермини-компьютеров. Появление компьютеров нового класса связано с успехами в технологии их изготовления, с расширением их функциональных возможностей. Так, увеличилась скорость, и расширился набор команд центрального процессора, выросла пропускная способность устройств ввода-вывода, расширен диапазон адресов, увеличился ассортимент внешних устройств.

Одновременно с этим появилась возможность создать для супермини-компьютеров богатое программное обеспечение, существенно отличающееся от математического обеспечения типичных миникомпьютеров. В то же время это математическое обеспечение ориентировано на интерактивный режим работы пользователя с машиной, т. е. на непосредственное общение. Крупными производителями программного обеспечения можно считеть: Microsoft, Symantec, Borland, Microprose и др.

В начале 70-х годов американская фирма Intel предложила вместо интегрального модуля с жесткой логикой разработать стандартный логический блок, конкретное назначение которого можно определить после его изготовления, т. е. создать программируемую интегральную схему. Так появился микропроцессор - многофункциональный цифровой микроэлектронный модуль с программируемой логикой, сделавший революцию в электронике и технике обработки информации.

Основой микропроцессора является большая интегральная схема, в которой происходят все функциональные и вычислительные операции. Микропроцессор составляет основу микрокомпьютера.

В 1980 г. основу почти всех микрокомпьютеров составляли однокристальные микропроцессоры, у которых на одном кристалле выполнены центральный процессор, устройство памяти, устройство ввода-вывода и другие логические схемы. Число транзисторов на одном кристалле достигало примерно 70 тыс. штук. В настоящее время уже имеются монокристаллические микросхемы с числом элементарных вентилей до 5 млн. штук, что закладывает основу для перехода к микрокомпьютерам следующего нового поколения.

Производство микрокомпьютеров происходит в нарастающем темпе.

К первому поколению персональных компьютеров американского производства относятся компьютер TRS-80 на базе микропроцессора Z80 фирмы Zilog, компьютер Apple - II фирмы Apple Computer, компьютер PET фирмы Commondore Int. на базе микропроцессора 6502 фирмы MOS Technology.

Ко второму поколению принадлежат компьютеры IBM Personal Computer фирмы IBM с микропроцессором i8088, Rainbow-100 фирмы DEC на базе микропроцессора Z80A/i8088.

Микрокомпьютеры второго поколения с 16-разрядным словом отличаются от своих предшественников включением в комплект кроме гибких дисков также малогабаритных жестких дисков большой емкости, графических устройств, увеличенным объемом оперативной памяти до 256 КБайт - 2 Мбайт. Остальные поколения являются производными от этих.

Характеристики персональных компьютеров постоянно улучшаются, и в настоящее время технические характеристики и области применения мини- и микрокомпьютеров перекрываются.

В конце 80-х годов существовала проблема отставания математического вооружения компьютера от его физических возможностей. Зато в настоящее время данная проблема преобразилась с точностью до наоборот. Т.е. современное программное обеспечение способно использовать все системные ресурсы компьютера и при этом требовать еще большего увеличения данных ресурсов.

Пусть, например, необходимо ввести данные в компьютер. Этот процесс следует начать с тестовых проверок, чтобы убедиться в физической возможности осуществления этой процедуры. Поэтому работа начинается с того, что устройство ввода посылает в компьютер сигнал, указывающий на готовность ввода порции информации. В свою очередь, компьютер должен известить устройство ввода, т. е. оператора, о том, что он закончил обработку предыдущей порции данных и готов к приему следующей. В результате происходящего обмена сигналами устанавливается режим, позволяющий вводить новую порцию данных.

При выводе информации из компьютера устройство вывода должно известить компьютер о том, что оно готово воспринимать данные, т. е. послать в компьютер так называемый "сигнал занятости". Компьютер начнет выдавать данные только после проверки наличия такого сигнала. В свою очередь, компьютер должен послать в устройство вывода сигнал готовности передавать данные и устройство вывода должно убедиться в наличии такого сигнала.

Этот очевидный режим обмена информацией между компьютером и периферийными устройствами имеет недостаток - нерациональное использование времени, так как компьютер значительную часть времени находится в режиме ожидания. Поэтому разработан более экономичный режим обмена данными - ввод и вывод по прерыванию. В этом режиме выполнение основной программы компьютером чередуется с выполнением подпрограмм ввода и вывода.

**Повышение производительности в компьютерных сетях.**

Быстрое проникновение информационных технологий в коммерцию, банковское дело, образование и сферу развлечений в совокупности с неуклонно увеличивающейся мощностью компьютеров и емкостью устройств хранения данных предъявляет все большие требования к сетям связи. На повестку дня выходят широкополосные каналы связи.

Согласно прогнозам, мощность персональных компьютеров и high-end-вычислительных машин в ближайшие 15 лет увеличится более чем в тысячу раз, соответственно и потребности в объеме трафика по опорным сетям связи вырастут в десятки и сотни раз. Эти тенденции стимулируют исследования в области высокопроизводительных сетевых решений. Самые быстрые линии передачи данных - оптоволоконные - требуют соответствующие технологии построения быстрых и гибких сетей. Одной из таких технологий является временное уплотнение каналов.

Современные технологии ATM и SDH решают задачу повышения производительности сетей лишь в краткосрочной перспективе. Их дальнейшее развитие сопровождается увеличением сложности и цены электронного оборудования, заставляя разработчиков обратить взор в сторону оптических технологий связи.

Связь по оптоволокну - пока единственный способ удовлетворить огромные потребности в скоростной передаче данных. Главными технологиями, позволяющими наиболее полно на сегодняшний день использовать поистине громадные возможности волокна, являются уплотнение каналов с разделением по спектру (WDM, или спектральное уплотнение) и с временным разделением (OTDM, или временное уплотнение).

И та и другая технологии важны не только для повышения скорости передачи данных, но и для ускорения коммутации и маршрутизации. Уже разработаны необходимые средства для решения этих задач без использования электронных устройств. Таким образом, снимаются ограничения, накладываемые электроникой. Временное и спектральное уплотнения не являются взаимоисключающими, хотя использование первого с некоторыми видами волокон затруднено. Более того, эти подходы могут быть скомбинированы. Небольшое число OTDM-каналов может быть объединено с помощью WDM, увеличивая емкость линии.

Представим, что нам нужно установить локальную сеть с большой пропускной способностью, так сказать, с заделом на будущее. Чем соединить между собой компьютеры? Еще недавно ответ был очевиден - коаксиал или витая пара (UTP) с третьей по пятую категории. Если речь шла о магистральных коммуникациях, прокладывалось оптическое стекловолокно. С развитием технологии выбор расширился. Появилась расширенная 5-я категория UTP, экранированная витая пара (так называемая 6-я европейская категория UTP). Снизилась стоимость оптоволокна. Кроме этого, на рынке кабельных соединений для локальных сетей возникло новое явление: пластиковое, или полимерное, оптическое волокно. По-английски оно называется Plastic Optical Fiber или сокращенно POF. Аналитики считают, что POF должно заполнить существующую нишу между медными проводами и стекловолокном по соотношению цена-производительность.

Пластиковое оптоволокно уже в течение нескольких лет производится в Японии и США. Оно употребляется для освещения (в качестве световодов), а также в робототехнике, промышленности и автомобилях. Число сетевых применений POF, впрочем, пока весьма невелико. Несколько американских компаний используют POF в качестве коротких (до 100 м) магистралей передачи данных. Новая разработка - POF с изменяющимся в поперечном сечении показателем преломления (graded-index POF) - обещает увеличить пропускную способность пластического волокна до 2 гигабит в секунду на дистанциях до 100 метров.

В каких отношениях находится POF со стеклом и медью с точки зрения технических характеристик и экономической выгоды? Пластиковые световоды способны работать в самых суровых температурных режимах, мыслимых для современных сетей, - от -40њ до 85њC. Без ущерба для оптических характеристик они могут выдерживать радиус изгиба до 20 мм и не ломаются даже при радиусе изгиба в 1 мм. Такая гибкость позволяет с легкостью прокладывать POF там же, где и медные провода, пропуская световод через стены и вентиляционные короба.

Область применения для POF оказывается весьма широкой. Помимо локальных сетей в их привычном понимании, пластиковые световоды могут использоваться в качестве сетевой основы для самолетов, компьютеризированных автомобилей, военного снаряжения и обмундирования, а также везде, где недопустимо возникновение электрических наводок и полей.

**Повышение производительности процессоров.**

Говорить о перспективах развития вычислительной техники довольно сложно. С одной стороны, мы привыкли к сумасшедшим, не имеющим аналогов в истории человечества темпам роста ее мощностей и проникновению в самые разнообразные сферы жизни человеческого общества С другой стороны, действительно, даже как-то не верится, что подобные темпы смогут сохранить такими же высокими.

К двухтысячному году микропроцессоры достигнут тактовой частоты 900 мегагерц, а их производительность по iSPEC95 будет равна примерно 60. К 2006 году частота подскочит до 4 ГГц, а производительность - до 500 (для сравнения: производительность сегодняшнего Pentium Pro, работающего на частоте 200 МГц, составляет 5,1). Рост производительности будет достигнут двумя путями: повышением частоты и степени параллелизма в архитектуре процессора.

Спрос на скорость и мощность будет расти по мере того, как в сегодняшние приложения начнут вводить видео, звук, анимацию, цвет, трехмерные изображения и другие средства, которые должны сделать PC и программы проще для использования.

**ММХ-революция в микропроцессорной эволюции.**

8 января 1997 года корпорация Intel анонсировала долгожданный процессор Pentium., выполненный с использованием технологии MMX.. Его официальное представление на территории СНГ и стран Балтии прошло 22 января в Москве, в киноцентре "Кодак Киномир". В этом материале представлены основные факты, сопутствующие появлению новой технологии от Intel.

Новый процессор, выпускаемый в четырех модификациях, представляет собой наиболее производительный из всех имеющихся на сегодня процессоров Pentium. Для настольных ПК, ориентированных на сегмент рынка потребительских товаров, процессоры выпускаются с рабочими частотами 166 и 200 МГц, а для hi-end ноутбуков разработаны процессоры с тактовыми частотами 150 и 166 МГц. На это обстоятельство наблюдатели обращают самое пристальное внимание - впервые за свою историю корпорация представляет "ноутбучные" процессоры одновременно с решениями для настольных систем.

**Технология.**

По мнению корпорации анонсированная в марте прошлого года технология MMX представляет собой наиболее существенное улучшение архитектуры процессоров Intel за последние 10 лет, с момента появления 32-х разрядного Intel386. Разработка этой технологии началась около пяти лет назад в ответ на быстрое развитие вычислительных систем, связанных с обработкой различных видов информации. Были проведены исследования широкого круга программ для обработки изображений, MPEG видео, синтеза музыки, сжатия речи и ее распознавания, игровых, ориентированных на видеоконференции и многих других. В них выделялись подпрограммы, наиболее активно эксплуатирующие вычислительные мощности. Затем, они были тщательно проанализированы. В результате такого анализа было выявлено то общее, что необходимо для эффективного выполнения различных категорий программ.

Основой MMX является архитектура SIMD (Single Instraction Multiply Data - "одна инструкция над многими данными"). Суть ее состоит в том, что данные поступают в процессор в виде 64-битных пакетов, которые обрабатываются одной командой. Кроме того логика процессора пополнилась 57 новыми инструкциями, повышающими производительность при выполнении наиболее типичных циклов, характерных для приложений, использующих большое количество аудиовизуальной информации.

Восемь 64-разрядных регистров MMX физически используют те же регистры, что и операции с плавающей точкой. С одной стороны подобный шаг обеспечивает полную совместимость с предшествующей архитектурой Intel и, как следствие, полную совместимость с широко используемыми операционными системами и прикладным программным обеспечением. С другой стороны на переключение с выполнения ММХ-инструкций, на действия с плавающей запятой процессору требуется 50 тактов, что рассматривается критиками как существенный минус архитектуры. Однако Intel с этим категорически не соглашается, утверждая, что подобное решение оптимально и удовлетворяет производителей как программного, так и аппаратного обеспечения.

Новые процессоры разработаны на основе созданной в Intel улучшенной 0,35-микронной технологии, которая позволяет получить более высокую производительность при меньшем потреблении мощности. Процессоры содержат по 4,5 млн. транзисторов и используют два уровня напряжения питания. Для совместимости с существующими компонентами процессор работают при напряжении 3,3 В, но ядро процессора в версии для настольных ПК работает при напряжении 2,8 В, а в версии для портативных ПК - при напряжении 2,45В. Работа ядра при пониженном напряжении позволяет снизить мощность рассеяния у новых процессоров до уровня их предшественников без технологии ММХ. При этом максимальное рассеяние мощности в случае с процессором, используемым в настольных ПК, составляет 15,7 Вт., а для процессора, используемого в портативных ПК - 7,8 Вт. Так что категорически не следует устанавливать ММХ-процессоры в старые, неприспособленные для них материнские платы! Эксперты полагают, что расплата наступит не сразу, но в итоге устойчиво процессоры работать не будут, при этом нагрев будет измеряться просто недопустимыми величинами, что в конце концов приведет к неминуемой аварии. Поэтому остается ждать процессоров Pentium Overdrive, выполненных по новой технологии и снабженных преобразователями напряжения, выпуск которых начнется уже в первой половине текущего года.

В соответствии с прогнозами специалистов Intel можно ожидать дальнейшего роста показателей микропроцессоров и по плотности интеграции, и по быстродействию. Но вполне возможно, что общие темпы повышения производительности компьютеров несколько снизятся.

Здесь хотелось бы отметить, что сравнение Pentium и Pentium Pro даже сегодня является несколько преждевременным. По сути дела, Pentium Pro был экспериментальной и не очень удачной моделью (например, идея интеграции на одном кристалле самого процессора кэш-памяти и их работы на одной тактовой частоте явно не оправдалась). На самом деле, фактически рабочая история семейства P6 начинается только сейчас, с выходом Pentium II: у него есть отличный потенциал для снижения себестоимости и одновременно - повышения частоты.

Возможное снижение темпов роста производительности микропроцессоров - явление довольно понятное. Дело в том, что с точки зрения большинства вычислительных задач размерность данных в 32 бита является оптимальной. 16-разрядный процессор работал существенно быстрее 8-разрядного, а 32-разрядный - и того быстрее. Но вот переход к 64-разрядной схеме уже вряд ли принесет столь же весомый результат. Вполне возможно, что будущий Merced будет действительно очень сильно опережать Pentium II, но только на 64-разрядных тестах.

**Основные проблемы**

Одна из самых фундаментальных проблем, которые предстоит решить: как справляться с растущей сложностью изделия и численностью команды разработчиков? Чтобы создать новое поколение продуктов, сегодня и в будущем сотни человек должны работать как одна команда.

Следующее препятствие: как гарантировать работоспособность и совместимость? Если проверять все вычислительные ситуации и все аспекты совместимости, количество комбинаций будет приближаться к бесконечности. Ясно, что нам потребуются качественно иные способы, чтобы справляться с микросхемами, содержащими 350 миллионов транзисторов.

Третья группа проблем связана с потребляемой мощностью. Напряжение питания придется понизить до одного вольта, для чего потребуется внести много нового в микроархитектуру, а также в процесс и программное обеспечение разработки.

Наконец, главным тормозом роста производительности будут внутрисхемные соединения - до тех пор пока не удастся открыть материалы с меньшими сопротивлением и емкостью. В современной микросхеме Pentium Pro содержится пять металлизированных слоев, благодаря чему сокращается расстояние между компонентами и сигналы передаются быстрее. В новом поколении процессоров этих слоев потребуется гораздо больше. История показывает, что технология металлизации развивается медленно. Чтобы сделать процессор 2006 года, надо срочно разворачивать новые исследования.

**Одно из перспективных решений.**

Одним из путей повышения производительности процессоров является новейшая технология производства микросхем на основе медных проводников.

Собственно, о прорыве в Cu-технологии изготовления микрочипов компания IBM заявила еще в сентябре 1997 года. Но за прошедшие до того пять лет напряженных лабораторных исследований и испытаний недостатка в подобных постоянно повторяющихся заявлениях уже никто не испытывал. И вот только последовавшие за этим октябрьские и ноябрьские события в мире большой полупроводниковой промышленности дали понять, что Великая медная революция, похоже, свершилась, поскольку вышла из лабораторий и теперь широко внедряется в серийном оборудовании полупроводниковой промышленности.

О преимуществах двойной дамасской медной технологии:Damascus Complete Copper, название технологической линейки оборудования альянса Lam и Novellus (и примкнувшей к ним IPEC), - перед традиционной алюминиевой. Попутно нужно заметить, что названный дамасским процесс имеет довольно слабое касательство к металлургии булатной стали. Похожего между ними маловато, кроме, возможно, удачно найденного гетерогенного сочетания различных медных структур. Гомогенные пленки меди просто никак не вписывались в существующие технологические приемы. Ну и, естественно, очень похожа секретность, которой окружены технологические режимы и параметры получения той и другой "дамасской" структуры.

Лучшая (чем у алюминия) проводимость меди позволяет исключить до 200 технологических операций (этапов) в изготовлении чипа. Это сильно, если вспомнить, что еще лет десять назад чип изготавливался за 60 операций, сегодня же производство микропроцессора требует 800 и более этапов.

Работая на одной тактовой частоте, чипы с медными межсоединениями будут потреблять на 30% меньше энергии, чем "классические". Учитывая же двух-трехкратное сокращение линейных размеров, достигаемое с помощью меди, подобная экономия выразится в еще больших значениях. Здесь уместно представить себе легкий и негорячий палмтоп (или хэнд-хелд - как кому нравится) с процессором, равным по мощности сегодняшнему Pentium II, только на частотах порядка 1 ГГц, с кэшем первого уровня под 1-2 Гбайт, с флэш-памятью под 300-500 Мбайт и/или RAM-диском 1-2 Гбайт.

Обычно средние затраты на этап технологического процесса в полупроводниковой промышленности имеют тенденцию снижаться на 25-30% каждый год. Внедрение двойного дамасского процесса на уровне внутренней разводки сократит общие затраты на 30% разом, сохраняя общую тенденцию сокращаться и далее, из года в год.

Для уровня 0,13 мкм и менее задержки в медных проводниках вдвое меньше, чем они могли бы быть в подобных (гипотетических, по большей части) структурах Al-SiO2.

Кроме того, в алюминиевых тонких (ширина около 0,25 мкм) проводниках плотность тока уже такова, что происходит электромиграция алюминия, приводящая к отказам. Лучшая сопротивляемость этому эффекту, характерная для меди, позволяет достаточно легко преодолеть предел по ширине проводника. Теперь остаются ограничения типа слишком высокой диэлектрической проницаемости у SiO2 (между слоями металла лежит именно этот материал). С преодолением этого недостатка и внедрением более совершенной литографии медь будет применяться до пределов 0,13 мкм.

Два основных отличия двойного дамасского медного процесса изготовления межсоединений от традиционной алюминиевой технологии состоят в следующем.

Во-первых, операцией, определяющей минимальную ширину и шаг разводки в случае Al, является травление металла, а планаризация (выравнивание обрабатываемой поверхности чипа по горизонтали) каждого металлического уровня осуществляется на этапах заполнения промежутка и CMP (химико-механической планаризации) диэлектрика. В процессе же изготовления медной разводки этапом, определяющим минимальную ширину и шаг проводников, является не травление металла, а более простое травление диэлектрика. Задачу планаризации выполняют на этапах осаждения и CMP меди.

Во-вторых, двойной дамасский процесс обладает еще одним преимуществом как перед обычной дамасской технологией, так и субтрактивным процессом, применяемым в настоящее время для изготовления алюминиевой разводки, он позволяет примерно на треть сократить число технологических этапов.

**Влияние архитектуры микропроцессоров на повышение производительности.**

В тяжёлой ситуации оказались производители микропроцессоров в конце девяностых годов. Сколько ни увеличивали они производительность процессоров, потребностей пользователей удовлетворить не могли. А остановиться - означало умереть: перестав крутить педали, упасть с велосипеда.

Наращивать тактовую частоту день ото дня становилось все труднее. Тогда разработчики пошли другим путем: оптимизировали исполнительные цепи, чтобы большинство команд исполнялось всего за один такт микропроцессора, ввели новые инструкции и векторные операции (технологии MMX и 3Dnow!)...

Сегодня можно с уверенностью сказать, что RISC- и CISC-архитектуры исчерпали себя, достигнув сопоставимой производительности. Но программисты, словно не заметив этого, все еще продолжают "утяжелять" программное обеспечение: Windows 2000 будет построена на объектах COM и COM+. С точки зрения разработчиков это хорошо, ибо позволит писать более устойчивый и свободный от ошибок программный код, но с точки зрения микропроцессора один только вызов объекта COM+ распадается на тысячи команд и очень-очень много тактов.

Без дальнейшего роста вычислительных мощностей внедрение этих технологий в повседневную жизнь просто немыслимо! Поэтому уже сегодня появляются многопроцессорные системы, ориентированные на домашние и офисные компьютеры.

Узким местом микропроцессоров традиционных архитектур стала выборка и декодирование инструкций. Действительно, в одном кристалле нетрудно разместить несколько независимых функциональных устройств, но только одно из них сможет обрабатывать поток команд. Почему? Очень просто: исполнять следующую инструкцию можно, только полностью уверившись, что ей не потребуется результат работы предыдущей.

**СУПЕРСКАЛЯРНАЯ АРХИТЕКТУРА**

Выходит, что исполнять за один такт можно и более одной инструкции? Действительно, что нам мешает синхронно исполнять нечто вроде:

MOV AX,1234h ; Записать в регистр AX число 1234h

MOV CX,DX ; Записать в регистр CX значение регистра DX

Достаточно лишь, чтобы устройство выборки инструкций позволяло декодировать обе команды за один такт. Для RISC с их фиксированной длиной команд это вообще не составляло никакой проблемы (подробнее - в статье "RISC vs. CISC").

Сложный набор инструкций CISC доставил немало головной боли разработчикам, но все же, ценой инженерных озарений и сложных аппаратных решений, были построены микропроцессоры, которые успевали декодировать две и более распространенные инструкции за один такт.

Словом, построение подобных декодеров не было непреодолимой преградой. Трудность заключалась в том, что далеко не все команды можно выполнять параллельно. Например:

MOV AX,1234h ; Записать в регистр AX число 1234h

ADD DX,AX ; Сложить содержимое регистра DX с регистром AX

Пока не будет известен результат работы первой команды, выполнение второй невозможно. Следовательно, микропроцессор будет простаивать, а пользователь пить кофе, созерцая на экране песочные часы.

По статистике только десять процентов смежных команд не используют результатов работы друг друга. Стоит ли мизерное увеличение производительности усложнения процессора? Оказывается, да: если немного подумать и еще чуточку усложнить анализ зависимости между командами.

В самом деле, если пример, приведенный выше, переписать как

MOV AX,1234h

ADD DX,1234h

мы получим идентичный результат, но обе команды могут быть исполнены параллельно всего за один такт процессора. Можно пойти дальше и задержать выполнение первой инструкции до той поры, пока значение регистра AX не потребуется в явном виде. Если оно и вовсе никогда не потребуется - мы сэкономим целый такт!

Идеи о подобной, на лету, оптимизации кода породили суперскалярные микропроцессоры, то есть такие, где параллелизм команд явно не указан и отслеживается процессором самостоятельно.

Однако Intel нашла подобные приемы оптимизации слишком трудными для реализации и сделала упор на изменение порядка выполнения команд. Так, последовательность

MOV AX,1234h ; Записать в регистр AX число 1234h

ADD DX,AX ; Сложить содержимое регистра DX с регистром AX

MOV CX,666h ; Записать в регистр CX число 666h

ADD BX.CX ; Сложить содержимое регистра BX с регистром CX

Можно исполнить в другом порядке:

MOV AX,1234h ; Записать в регистр AX число 1234h

MOV CX,666h ; Записать в регистр CX число 666h

ADD DX,AX ; Сложить содержимое регистра DX с регистром AX

ADD BX.CX ; Сложить содержимое регистра BX с регистром CX

Теперь соседние инструкции независимы и могут быть исполнены параллельно. Следовательно, приведенный выше пример может быть исполнен за два такта вместо четырех. Очень неплохой путь повышения производительности, но, к сожалению, очевидно тупиковый: усовершенствовав интеллектуальный "движок", можно найти способ параллельного исполнения четырех команд, но сомнительно, чтобы существовал волшебный способ устранения зависимости между восемью и более командами.

RISC в этой ситуации оказались в более выигрышном положении. Ограниченный набор регистров CISC порождал проблемы аналогично следующей:

MOV AX,1234h ; Записать в регистр AX число 1234h

ADD DX,AX ; Сложить содержимое регистра DX с регистром AX

MOV AX,666h ; Записать в регистр CX число 666h

ADD BX.AX ; Сложить содержимое регистра BX с регистром AX

Теперь уже невозможно одновременно выполнить первую и третью строки, однако этой, казалось бы, на первый взгляд, неразрешимой проблеме быстро было найдено красивое решение. В действительности зависимость между двумя командами ложная. Очевидно (даже машине), что должны использоваться дополнительные регистры. Но что нас ограничивает? Давайте переименуем регистры в AX~1 и AX~2 соответственно. Тогда получим следующий код:

MOV AX~1,1234h ; Записать в регистр AX число 1234h

ADD DX, AX~1 ; Сложить содержимое регистра DX с регистром AX

MOV AX~2,666h ; Записать в регистр CX число 666h

ADD BX, AX~2 ; Сложить содержимое регистра BX с регистром AX

Разумеется, теперь никаких проблем с параллельным исполнением уже не возникнет. Конечно же, потребуется больше регистров! Но регистры дешевы (всего лишь набор триггеров из 4-6 транзисторов), а за быстродействие потребитель деньги охотно заплатит.

Проблема в том, что добавление новых регистров потребует перекомпиляции всего существующего программного обеспечения и изменения адресации. Одним словом, приведет к несовместимости с предыдущими моделями. Для кого-то это, может, было и не критично, но только не для Intel, которой обратная совместимость нужна обязательно.

И Intel находит блестящее решение. Существующие регистры при декодировании команды проецируются на гораздо больший набор внутренних. И как только обнаруживается ложная зависимость, очередной регистр переименовывается. Разрыв в производительности между CISC и RISC снова утерян.

**ПАРАЛЛЕЛИЗМ В МИКРОПРОЦЕССОРАХ RISC и CISC**

**Микропроцессоры PowerPC (RISC)**

Процессор PowerPC был разработан в результате тесного сотрудничества ведущих производителей индустрии - IBM, Apple и Motorola. Он воплотил в себе гений тысяч инженеров, долгое время был лидером среди собратьев в отношении цена/производительность.

Суперскалярное RISC-ядро позволяло за один такт выполнять до четырех команд благодаря четырем конвейерам выборки и шести независимым функциональным устройствам: трем целочисленным АЛУ, блоку вещественной арифметики (обрабатывающему числа с плавающей точкой), модулю чтения\записи результатов и блоку переходов. Для параллельного исполнения выборка инструкций должна соответствовать набору имеющихся устройств. Четыре целочисленные операции за один такт не выполнятся, а вот три целочисленные и одна вещественная могут.

Для повышения производительности и поддержки эффективной загрузки функциональных блоков в PowerPC используется динамическое предсказание условных переходов вкупе с упреждающим выполнением кода на глубину до четырех предсказанных переходов.

При этом результаты работы команд записываются в промежуточный кэш-буфер, который обнуляется в случае ошибочного предсказания. Что представляет собой динамическое предсказание ветвлений? В микропроцессорах PowerPC использован простой и надежный механизм таблицы предыстории переходов (или, говоря красивым техническим языком, BTH - Branch History Table).

В кэш этой таблицы заносятся адреса и результаты ветвления. Если попадается условный переход, который уже был занесен в таблицу, процессор полагает, что результат ветвления окажется тот же самый, и начинает упреждающее выполнение соответствующей ветви.

Этот механизм позволяет угадывать направление перехода в среднем в 95 случаях из 100, что значительно выше, чем у CISC-микропроцессоров. Впрочем, и самих условных переходов в RISC'ах значительно больше, да и находятся они большей частью в циклах, тогда как в CISC они иррегулярно разбросаны по всему коду и имеют тесную связь между обрабатываемыми в настоящий момент данными и направлением ветвления. Попросту говоря, они менее периодичны и предсказуемы. Но ввиду того, что самих условных переходов значительно меньше, удачность или неудачность предсказаний незначительно отражается на общей производительности.

PowerPC активно использует технологию переупорядочения последовательности выполнения команд. В любой момент могут быть выполнены четыре произвольные инструкции из шестнадцатиэлементного буфера команд.

**Микропроцессор Pentium Pro**

Высокая производительность была достигнута в большой степени благодаря улучшению внутреннего параллелизма архитектуры процессора. Прежде всего, это разнесенная архитектура и динамическое исполнение команд.

Последнее представляет собой опережающее исполнение команд, предсказание переходов и переупорядочение очереди выполнения инструкций. Кроме того, в Pentium широко используется нехарактерная для RISC технология переименования регистров для устранения ложной зависимости (в RISC просто слишком много регистров, чтобы в этом возникла существенная необходимость или оказалось недостаточно переупорядочения команд).

Конвейеров у Pentium Pro два. Однако это еще не означает, что Pentium способен выполнить две любые инструкции параллельно. Исходя из сложности команд CISC и того факта, что каждая инструкция может задействовать произвольное (читай: временами очень большое) число функциональных устройств микропроцессора, их все пришлось бы дублировать. В результате далеко не всякие инструкции выполняются параллельно.

Для борьбы с иррегулярными переходами в Pentium был использован значительно более сложный по сравнению с RISC-системами статически-динамический анализатор условия ветвления вместе с детектором вложенных конструкций (например, вложенных циклов). Но даже этот механизм не позволил достичь свыше 80 процентов угадываний.

Видно, что у RISC-микропроцессоров параллелизм выражен в большей степени, чем у CISC. Фиксированная (или близкая к ней) длина инструкций облегчила разработку многоконвейерных реализаций. Простые инструкции по отдельности задействовали лишь одно функциональное устройство. Например, типичная CISC-операция:

ADD AX, [SI+BX+66h]

Необходимо разобрать сложное поле адресации, чтобы понять, что это именно "SI+BX+66h", а не что-то другое; сложить (то есть вычислить этот результат), затем загрузить содержимое требуемой ячейки и, наконец, опять сложить его со значением регистра AX.

Следовательно, для выполнения инструкции за один такт требуется два арифметических устройства и один загрузчик. Но и в этом случае все действия должны выполняться последовательно. То есть ВЫЧИСЛЕНИЕ АДРЕСА, ЗАГРУЗКА, ВЫЧИСЛЕНИЕ РЕЗУЛЬТАТА. И никакие действия не могут быть спарены. Хоть умри, но потребуется три такта на выполнение, или хотя бы одно из устройств должно функционировать на удвоенной тактовой частоте.

RISC казались более приятным полигоном для проникновения параллельных вычислений в бытовые компьютеры. А в том, что будущее принадлежит параллельным архитектурам, уже никто не сомневался: линейные (последовательные) вычисления наткнулись на непреодолимые на сегодняшнем этапе физические и технические ограничения и к середине девяностых сохранились только там, где производительность была некритична (скажем, микрокалькуляторы, наручные часы, аппаратура определения номера и др.).

**ВЫБОР ДРУГОГО ПУТИ**

Производители CISC-архитектур недолго оставались в тупике. Они решили усилить параллелизм не в исполнении команд, а в обработке данных, введя поддержку векторных операций. Именно в этом и заключается широко известная MMX-технология.

Однако новая технология не давала никакого выигрыша на старых приложениях - не спасала даже перекомпиляция. Существенно повлиять на быстродействие могло только принципиальное изменение идеологии программирования: переход на векторные операции.

Несомненный плюс - явный параллелизм всего в одной инструкции там, где раньше их требовалась сотня. Использование MMX дает до 60 процентов прироста производительности приложений, манипулирующих графикой, звуком или аналогичными вычислениями и обработкой больших массивов чисел.

AMD не отстала от конкурентов и активно продвигает на рынок свой вариант реализации MMX - технологию 3DNow!, ориентированную на вычисления, связанные с трехмерной графикой.

Однако существовали многие ограничения на внедрение и распространение этих технологий. Прежде всего, векторные вычисления не типичны для платформы PC. Они требуют не столько поддержки со стороны компиляторов (компиляторам что - просто новый набор инструкций), сколько именно используемых программистами алгоритмов и языков. Ни Си, ни Паскаль, ни другие популярные языки не ориентированы на векторные операции.

Более того, не ориентированы на них и задачи, решаемые персональным компьютером. Большую часть времени процессор занят не наукоемкими вычислениями, а "работой по хозяйству": обмен с периферией, прорисовка окон, бесконечные вызовы функций и передача многочисленных параметров с постоянным контролем границ последних.

Во всех вышеперечисленных операциях операндами выступают именно скаляры, никак не векторы. Конечно, графические приложения ускоряют работу, но не в той степени, как специализированные графические процессоры. В целом польза от MMX оказалась настолько сомнительной, что сегодня только единичные приложения используют расширенный набор команд.

Гораздо большего можно было добиться, заставив компилятор располагать инструкции так, чтобы максимально увеличить возможность спаривания и распараллеливания вычислений. Иными словами, помогать процессору в оптимизации. Прежде всего, не использовать сложную адресацию и ограничить себя только командами, которые выполняются за один такт, при этом - не мешая выполнению других.

Но не приходим ли мы добровольно к RISC-концепции, вновь возвращаясь к началу на очередном витке прогресса? К тому же правила оптимизации даже у соседних моделей процессоров (например, Pentium 75, Pentium 100, Pentium 166 MMX, Pentium II) отличаются очень и очень. Настолько, что код, оптимизированный под один процессор, может исполняться медленнее неоптимизированного - на другом.

Воистину, "мартышкин труд". Это что же, при появлении нового микропроцессора перекомпилировать все программное обеспечение заново? Поставлять покупателю исходные тексты с компилятором? Или пытаться в ущерб производительности найти компромиссный вариант? Причем любое решение было бы надругательством над самой концепцией суперскалярной архитектуры: параллелизм распознает сам процессор, и ему не требуется помощи со стороны компилятора.

Вот если бы был волшебный способ не подстраиваться под логику работы процессора, а явно указать ему связи между командами... Именно так и возникла архитектура VLIW.

VLIW (процессоры с длинным командным словом)

Параллелизм был свойствен еще первым ламповым компьютерам. Тогда это был единственный путь увеличения производительности. Потом, с совершенствованием технологий, о нем то забывали, то, когда вставали технические препятствия наращивания производительности, вспоминали.

Почему? Да потому, что параллелизм - это... как бы помягче выразиться? Нет, не плохо, скорее всего - излишне сложно и непривычно, и сопряжено с трудностями в разработке компиляторов.

Однако на рубеже второго тысячелетия компьютерная индустрия завела себя в тупик. Считается, что с поправкой на ионный ветер и длину волны излучателя, который рисует матрицу будущего кристалла, вкупе с некоторыми другими эффектами, невозможно (читай: технически неоправданно сложно) перейти порог 0,1 мкм в ближайшие десять лет, а выпускаемые сейчас по 0,18-мкм технологии микропроцессоры его уже почти достигли. (Texas Instruments уже разработала технологический процесс с проектными нормами 0,07 мкм. - Г.Б.)

Так же невозможно уменьшение до бесконечности напряжения питания. Во всяком случае, оно должно превышать фоновый "шум", вызванный случайными перескоками электронов, космическим излучением и электромагнитными наводками.

Менее проблематичным выглядит увеличение слоев металлизации. Технически это нетрудно, но вот экономически - неоправданно. Вероятность брака растет экспоненциально. Именно этим (высоким процентом брака) в первую очередь и объясняется значительная стоимость процессора Pentium Pro.

Это был тупик. Тупик, в который попали и CISC, и RISC. Ни одна из них не была приспособлена для параллельных вычислений, а суперскалярные архитектуры уже не обеспечивали должной производительности.

Архитектура VLIW была известна и ранее, хотя процессоры на ее основе были мало распространены. Наиболее известные модели были выпущены ныне уже не существующей фирмой Multiflow Computer. [6] В нашей стране по сходной архитектуре был построен суперкомпьютер "Эльбрус-3".

Из микропроцессоров, построенных по VLIW-архитектуре, можно назвать только семейство сигнальных чипов TMS320C6x фирмы Texas Instruments. Эта архитектура не стала бы знаменитой, если бы именно на нее не обратили внимание компании HP и Intel при разработке микропроцессора нового поколения Merced. Поэтому есть смысл рассмотреть ее подробнее.

Основной идеей VLIW был явный параллелизм, задаваемый на уровне команд. То есть в каждой команде имелось поле, в котором указывалась связь между инструкциями. Как правило, три команды объединялись в одно слово, в котором были возможны следующие варианты исполнения:

- все три команды выполняются параллельно;

- сначала выполняются первые две команды, потом третья;

- выполняется одна команда, а затем параллельно последующие две.

Изюминка в том, что сами слова можно было упаковывать по три (или более - это зависело только от конкретной реализации), и поскольку явно указана зависимость между командами, возможно неограниченно наращивать число параллельно выполняемых инструкций. Причем код мог без перекомпиляции эффективно работать на совместимых процессорах с разным числом исполняющих устройств. Хотя обычно ограничивались девятью. То есть три слова по три команды в каждом исполнялись за один такт.

Неплохой результат, верно? Особенно если вспомнить, что RISC были способны выполнить не более четырех команд, а CISC - и вовсе двух, тогда как для VLIW девять (или 16 = 4 по 4) инструкций не предел. Более того: если суперскалярные архитектуры только некоторые - удачные - команды исполняли параллельно, то VLIW - в большинстве случаев. Поэтому производительность может повышаться в десятки раз при незначительном аппаратном усложнении конструкции.

В чем же тогда камень преткновения? Почему архитектура так и не вышла из стен лабораторий и процессоров суперкомпьютеров, если технически она была достижима еще вчера?

Вся сложность в компиляторах. Точнее - в распараллеливании инструкций. Чтобы параллельно исполнять девять и более команд, необходимо расположить их так, чтобы не было явной зависимости. Теперь процессор не будет нам ни в чем помогать. Он будет лишь буква в букву следовать распоряжениям компилятора.

Сложность разработки качественных компиляторов значительно превышает сложность разработки самих процессоров. Кроме того, до сих пор нет ни одного массового языка, ориентированного на параллельные вычисления. Следовательно, если откинуть мысль о разработке нового языка как нереальную, придется по крайней мере вносить новшества в существующие языки.

Впрочем, у фирмы Texas Instruments имеется достойный компилятор, который разрабатывался очень длительное время и в настоящий момент близок если не к совершенству, то по крайней мере к товарному состоянию. Однако хороший результат можно получить только при тесном взаимодействии компилятора и программиста, в противном случае производительность сгенерированного кода будет незначительно превосходить традиционную суперскалярную. Это и есть камень преткновения на пути к широкому и быстрому внедрению VLIW-процессоров.

Merced будет выполнен по VLIW-архитектуре в духе RISC-машин. Но вспомним, что RISC - это идеализированная архитектура. И поэтому заимствуется не она сама, а лишь часть характерных для нее технологий.

Это прежде всего простая адресация и фиксированный размер операндов. Только так можно обеспечить высокоскоростную выборку инструкций для параллельного их исполнения. А компактность кода сегодня уже мало кого волнует. Ведь даже размер кэша первого уровня в некоторых моделях доходит до полутора мегабайт. Это больше, чем когда-то - оперативной памяти!

Другой характерной для RISC чертой будет разделение инструкций по категориям: чтения/записи в память и вычислительных операций. Это облегчит распараллеливание вычислений. Разумеется, и набор регистров обещает быть солидным. В том же Merced одних только целочисленных регистров планируется 128!

Но, несомненно, есть и типичные CISC-черты: прежде всего, набор команд. Аппаратная сложность процессора уже не преграда, поэтому команд он будет поддерживать столько, на сколько у разработчиков хватит фантазии. Не обойдут стороной и векторные операции, и богатый набор специализированных инструкций наподобие ряда Фурье и других.

Впрочем, все же новая архитектура больше наследует от RISC, чем от CISC. И реализуется на базе современных RISC-микропроцессоров. Однако Intel не была бы сама собой, если бы отказалась от совместимости с серией процессоров 80x86.

Таким образом, современная реализация VLIW-архитектуры наследует концепции и RISC, и CISC.

**Список литературы.**

1. http://www.intel.ru/contents/press/index.htm.
2. http://www.intel.com/th/thai/channel/report/page5.htm
3. Еженедельник «Компьютерра» (1998-1999)