

ГЛУБОКОВОДНОЕ ПОГРУЖЕНИЕ В ЧИПСЕТ INTEL 875P



*...чтобы оценить свежесть осетрины, вовсе не обязательно съедать всего осетра целиком.
Святослав Логинов*

Как-то, читая очередной компьютерный журнал, я задумался: а для кого, собственно, предназначены все эти обзоры чипсетов и прочих компьютерных комплектующих? Сравнительные характеристики и тесты производительности – это, бесспорно, нужно, полезно и хорошо, но вот за счет чего та или иная производительность в каждом конкретном случае достигается? Благодаря каким именно инженерным решениям ведущие компании «опускают» своих многочисленных конкурентов? А сами конкуренты – где они допустили просчет? Увы! Подавляющее большинство публикаций не дает ответа на этот вопрос. В результате философия архитектуры чипсета остается за кадром и весь разговор сводится к известному «вчера были большие, но по пять, а сегодня по три, но маленькие». Ни принципы функционирования чипсета, ни сопутствующая ему терминология при этом вообще не поясняются и у читателя складывается выхолощенный образ аппаратуры, на которой он работает. Помните анекдот: «Вы нарушили закон, и за это мы вас расстреляем! Какой, спрашиваете, закон? Этого мы вам сказать не можем, ведь этот закон – секретный.»? Шутка шуткою, но уж очень хорошо она иллюстрирует современную ситуацию: «Данная модель имеет ряд патентованных технологий Super X, Super Y и Double-Super Z, обеспечивающих ей наивысшую производительность. В чем заключается их суть – не спрашивайте! Не можем же мы раскрывать секрет фирмы первому встречному, верно?».

Вам еще не надоело покупать кота в мешке? Хотите заглянуть внутрь кристалла чипсета, разобраться, чем он живет, как дышит, насколько корректно работает с остальными устройствами? Тогда эта статья для вас!

КРИС КАСПЕРСКИ

Что такое чипсет, и какую роль в компьютерах типа IBM PC он играет? Читатели, собиравшие (или хотя бы разбиравшие) в свое время бессмертные ZX-Spectrum, наверняка помнят большое количество микросхем, живописно разбросанных по системной плате. Среди них были как микросхемы поддержки центрального процессора, так и контроллеры различных устройств (клавиатуры, памяти, экрана и т. д.). В совокупности они образовали набор системной логики, так называемый чипсет, дословно переводимый на русский язык как «набор микросхем». В фирменных Spectrum микросхема чипсета была всего одна, но разработанная по специальному заказу, она вмещала в себя все остальные. Таким образом, ничего таинственного в чипсете и нет – просто за счет высокой степени интеграции большое количество сопутствующих процессору устройств удается разместить на одном кристалле.

Собственно, чипсет первых IBM PC XT/AT отличался от чипсета Spectrum лишь тем, что имел контроллер шины ISA на борту, плюс контроллер DMA (устройства прямого доступа к памяти), плюс еще несколько контроллеров. Ни дискового контроллера, ни даже контроллера COM- и LPT-портов в первых IBM-овских чипсетах и в помине не было, а соответствующая функциональность обеспечивалась установкой специальной карты в слот расширения. Идея открытой архитектуры IBM PC как раз и заключалась в максимальном упрощении оснастки материнской платы за счет выноса всех остальных компонентов во внешние контроллеры.

Однако стремительное увеличение степени интеграции привело к тому, что вмонтировать в уже готовый кристалл пару-тройку лишних контроллеров стало намного дешевле, чем размещать эти же самые контроллеры на отдельной печатной плате. Так появились компьютеры, несущие на своем борту контроллеры гибких/жестких дисков, интегрированный звук/видео и... еще много всякой всячины! Количество устройств, заключенных в одной микросхеме, стремительно нарастало, и в какой-то момент разработчики решили разделить ее на две, получивших название северного и южного мостов (в другой терминологии – хабов).

Северный мост (названный так за свое традиционное расположение на чертежах) берет на себя наиболее ответственные задачи, требующие наивысшей пропускной способности, а потому и геометрической близости к процессору. Это контроллер системной шины, контроллер памяти, контроллер порта AGP, PCI-контроллер или контроллер внутренней шины для общения с южным мостом. В настоящее время пропускная способность периферийных устройств, «подцепленных» к южному мосту, превысила предельно допустимую пропускную способность шины PCI (133 Мб/сек) и производители, скрепя сердце, вынесли контроллер PCI-шины на южный мост, ну а в сам северный мост внедрили высокоскоростной контроллер внутренней шины (в Intel 82875P – это Hub Interface 1.5 с пропускной способностью в 266 Мб/сек, что вдвое выше, чем у штатной PCI. Впрочем, это еще не предел: MuTIOL-контроллер чипсета SiS 655 обеспечивает пропускную способность аж в 1,2 Гб/сек, то есть вчетверо быстрее Intel, правда, в подавляющем большинстве случаев эта возможность так и остается невостребованной, поскольку у вас попросту не найдется таких «прожорливых» периферийных устройств!).

Южный мост отвечает за ввод/вывод и включает в себя контроллер DMA, контроллер прерываний, контроллер таймера, контроллеры жестких и гибких дисков, контроллеры последовательных, параллельных и USB-портов и т. д., и т. п. Легко видеть, что производительность системы определяется фактически одним северным мостом, причем в силу значительных технологических и инженерных сложностей работы на таких запредельных скоростях именно на северный мост и приходится наибольшая доля архитектурных различий в конструкции чипсетов. Южные же мосты второстепенны, скучны, неповоротливы и неинтересны. Тот же Intel 875P в качестве южного моста использует допотопную микросхему Intel 82801EB, практически не изменившуюся с апреля 1999 года. Вы эту вялость жизни применительно к северным мостам представляете? Северные мосты – наиболее бурно и динамично развивающийся компонент компьютера, уступающий в интенсивности накала страстей разве что самим центральным процессорам (которые, собственно, северный мост и обслуживают). Это передовое направление техники и науки, и здесь нам есть чему поучиться! Данная статья целиком и полностью отдает себя именно северным мостам, ну а про южные вы читаете где-нибудь в другом месте.

Начнем с главного: вопреки распространенному заблуждению процессор взаимодействует с оперативной памятью не напрямую, а через специальный контроллер, подключенный к системной шине процессора приблизительно так же, как и остальные контроллеры периферийных устройств. Причем механизм обращения к портам ввода/вывода и к ячейкам оперативной памяти с точки зрения процессора практически идентичен. Процессор сначала выставляет на адресную шину требуемый адрес и в следующем такте уточняет тип запроса: происходит ли обращение к памяти, портам ввода/вывода или подтверждение прерывания. В некотором смысле оперативную память можно рассматривать как совокупность регистров ввода/вывода, каждый из которых хранит некоторое значение.

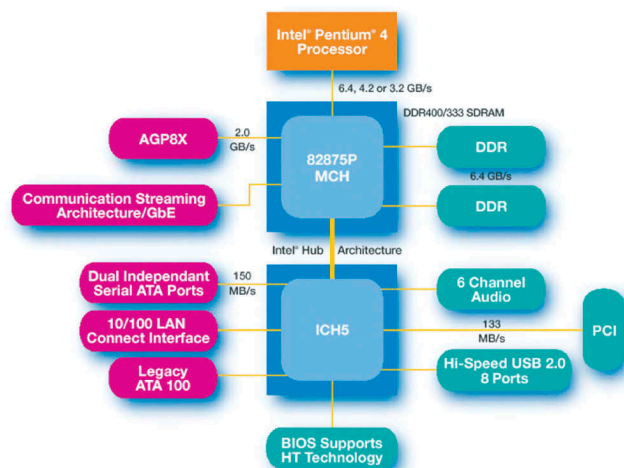
Механизм взаимодействия памяти и процессора выглядит приблизительно так: когда процессору требуется получить содержимое ячейки оперативной памяти, он, дождавшись освобождения шины, через механизм арбитража захватывает шину в свое владение (что занимает один такт) и в следующем такте передает адрес искомой

ячейки. Еще один такт уходит на уточнение типа запроса, назначение уникального идентификатора транзакции, сообщение длины запроса и маскировку байтов шины. Подробнее об этом можно прочитать в спецификациях на шины P6 и EV6, здесь же достаточно отметить, что эта фаза запроса осуществляется за три такта системной шины. Независимо от размера читаемой ячейки (байт, слово, двойное слово) длина запроса всегда равна размеру линейки L2-кэша, что составляет 32 байта для процессоров K6/P-II/P-III, 64 байта – для AMD Athlon и 128 байт – для P-4¹. Такое решение значительно увеличивает производительность памяти при последовательном чтении ячеек и практически не уменьшает ее при чтении ячеек вразброс, что и неудивительно, т.к. латентность чипсета в несколько раз превышает реальное время передачи данных и им можно пренебречь.

Контроллер шины (BIU – Bus Interface Unit), «вживленный» в северный мост чипсета, получив запрос от процессора, в зависимости от ситуации либо передает его соответствующему агенту (в нашем случае – контроллеру памяти), либо ставит запрос в очередь, если агент в этот момент чем-то занят. Потребность в очереди объясняется тем, что процессор может посылать очередной запрос, не дожидаясь завершения обработки предыдущего, а раз так – запросы приходится где-то хранить. Чем больше запросов может накапливать чипсет, тем выше максимально достижимый параллелизм обработки данных, а значит, выше и скорость. Чипсеты старого поколения (Intel 815, в частности) могли накапливать всего лишь до четырех запросов, однако с ростом отношения латентность/пропускная способность оперативной памяти размера очереди стало катастрофически не хватать и в Intel 875P/SiS 655 она была увеличена до 12 элементов.

При первой же возможности чипсет извлекает запрос из очереди и передает его контроллеру памяти (MCT – Memory Controller). В течение одного такта он декодирует полученный адрес в физический номер строки/столбца ячейки и передает его модулю памяти по сценарию, описанному в приложении «приблизительная схема взаимодействия памяти и процессора».

В зависимости от архитектуры контроллера памяти он работает с памятью либо на частоте системной шины (синхронный контроллер), либо поддерживает память любой другой частоты (асинхронный контроллер). Синхронные контроллеры ограничивают пользователей ПК в выборе модулей памяти, но, с другой стороны, асинхронные контроллеры менее производительны. Почему? Во-первых, в силу несоответствия частот, читаемые данные не могут быть непосредственно переданы на контроллер шины, и их приходится сначала складывать в промежуточный буфер, откуда шинный контроллер сможет их извлекать с нужной ему скоростью. (Аналогичная ситуация наблюдается и с записью). Во-вторых, если частота системной шины и частота памяти не соотносятся как целые числа, то перед началом обмена приходится дожидаться завершения текущего тактового импульса. Таких задержек (в просторечии пенальти) возникает две: одна – при передаче микросхеме памяти адреса требуемой ячейки, вторая – при передаче считанных данных шинному контроллеру. Все это



значительно увеличивает латентность подсистемы памяти, т. е. промежуток времени с момента посылки запроса до получения данных. SiS 655 – асинхронный контроллер со всеми вытекающими отсюда достоинствами и недостатками, а Intel 875P – это асинхронный контроллер, при совпадении частот шины и памяти автоматически переходящий в синхронный режим. Сочетая сильные стороны обоих типов контроллеров, чипсет Intel 875P практически не имеет недостатков (одним выстрелом – двух зайцев).

Другой фундаментальной характеристикой чипсета являются типы поддерживаемых им модулей памяти, что играет решающую роль в выборе чипсета пользователем (точнее, разработчиком материнской платы, но это не суть важно). Чипсет Intel 860, сделавший в свое время ставку на мало популярную память типа RDRAM, несмотря на все маркетинговые усилия, так и не получил большого распространения, поэтому компании Intel пришлось переориентироваться на DDR-SDRAM, де-факто ставшей массовой памятью к настоящему времени. Чипсет Intel 875P поддерживает три наиболее перспективных типа памяти: DDR266, DDR333 и DDR400, что выгодно отличает его от чипсета SiS 655, который память типа DDR400, увы, не поддерживает (подробную информацию о поддерживаемых типах памяти вы найдете в приложении «выбор и конфигурирование памяти»!).

Однако обеспечить формальную поддержку высокоскоростных типов памяти – это только полдела! Ведь с этой самой памятью еще надо уметь эффективно работать. Можно с сумасшедшей скоростью черпать воду наперсток, а можно степенно и деловито хлебать ее литровой пивной кружкой. Так что не надо путать тактовую частоту чего бы то ни было с реально достижимой производительностью.

Начнем с того, что пропускная способность наилучшей на сегодняшний день памяти DDR400, составляющая, как известно, 3,2 ГГб/сек, много меньше пропускной способности системной шины последних моделей Pentium-4, вмещающих в себя 4,3 ГГб/сек и 6,4 ГГб/сек на частотах в 4x133 МГц и 4x200 МГц соответственно. Причем отметим, что 3,2 ГГб/сек – это пиковая производительность, достижимая лишь при параллельной обработке данных, а на практике из-за высокой латентности чипсета даже при операции копирования блоков памяти она (пропускная способность) оказывается в полтора-два раза меньше! К тому же часть пропускной способности неизбежно «съедает» AGP-карта, и процессорной шине практически ничего не остается. Так какой же тогда смысл покупать (и выпускать!) процессоры с быстрыми шинами, если этой быстротой воспользоваться все равно не удастся?!

А поскольку надеяться на быстрый прогресс в области совершенствования памяти нам, увы, не приходится, разработчики были вынуждены прибегнуть к усилению параллелизма. Короче говоря, если один землекоп выкапывает один метр траншеи за час, то N землекопов выкопают N метров за это же самое время. Чипсет Intel 875P отличается от своих предшественников тем, что умеет работать с двумя парами модулей памяти параллельно. То есть если тот же Intel 845 загружал запрошенную у него кэш-строку за два пакетных цикла, то Intel 875P загружает ее за один, обращаясь сразу к двум модулям памяти одновременно! Эта мера вдвое увеличивает пропускную способность, но никак не

влияет на латентность! А поскольку для многих вычислительных алгоритмов величина латентности как раз и является доминирующей, то оптимизм по поводу такого параллелизма выходит не столь уж и радужным. При том же копировании производительность увеличивается не в двое, а всего в полтора раза! Таким образом, *на реальных задачах расчетная пропускная способность все равно не достигается!* Что же касается хаотичного доступа к памяти, характерного для таких структур, как деревья и списки (а это базовые структуры данных!), здесь ситуация еще хуже! Оперативная память отнюдь не так однородна, как кажется: она делится на банки, банки делятся на страницы, страницы делятся на строки, ну а строки делятся на ячейки. Прежде чем микросхема памяти «позволит» обратиться к ячейкам той или иной строки, соответствующая ей страница должна быть открыта, а на эту операцию требуется затратить весьма внушительное количество времени, практически сопоставимое с выполнением всего пакетного цикла целиком! Зато все запросы, направленные к открытой странице, могут выполняться без каких-либо задержек. Большинство модулей памяти имеют от четырех до восьми банков, а потому чипсет Intel 875P может удерживать в открытом состоянии до 32 страниц одновременно, что соответствует 256 Кб памяти. От максимально возможного количества открытых страниц напрямую зависит и производительность приложений, обрабатывающих более одного потока данных параллельно (к этой категории относятся, в частности, многие графические приложения). Отсюда: лучше иметь четыре отдельных модуля по 128 Мб, чем один на 512 Мб! При работе в двухканальном режиме это даст 16 одновременно открытых страниц плюс учетверенную (реально удвоенную) пропускную способность за счет усиленного параллелизма. Емкость и организация всех модулей при этом должна быть строго идентичной, в противном случае в многоканальном режиме чипсет просто не сможет работать! Требования к рабочей частоте менее жестки (более быстрая память будет работать на частоте самой медленной) и двухканальный режим разночастотные модули поддерживать в принципе будут, однако чипсет сможет работать только в обычном режиме адресации. А одна из главных вкусоностей Intel 875P как раз и заключается в том, что он поддерживает продвинутый режим динамической адресации.

Что же это за гусь такой и с чем его едят? В режиме обычной адресации (в котором функционирует подавляющее большинство чипсетов) банки памяти следуют один за другим и при последовательном чтении памяти располагаются так: страница 0 банк 0 → страница 0 банк 1 → страница 0 банк 2 → страница 0 банк 3 → страница 1 банк 1... Как следствие увеличивается удельная плотность банков на страницу (а значит, и возрастают накладные расходы на их более частое открытие/закрытие). В режиме динамической адресации чипсет отображает физические адреса памяти на адреса системной шины так, что банки начинают переключаться вдвое реже. Сначала следует последовательность страница 0 банк 0 → страница 0 банк 1 → страница 1 банк 1 → страница 1 банк 1 → страница 2 банк 0, а когда станицы первых двух банков полностью исчерпают себя: страница 0 банк 2 → страница 0 банк 3 → страница 1 банк 2 → стра-

ница 1 банк 3 → страница 2 банк 2. Достоинством нового решения является значительное уменьшение латентности чтения, что немаловажно для подавляющего большинства приложений и потому отказываться от этой возможности, право же, не стоит!

С другой вкусностью чипсета Intel 875P, так называемой PAT (Performance Acceleration Technology – технология увеличения производительности), ситуация скорее запутанна, чем ясна. В документации сказано лишь то, что эта технология увеличивает производительность за счет сокращения латентности чипсета, которая теперь (если верить документации) составляет всего два такта. На удивление малая величина, если принять во внимание длину цепочки и количество участников обработки запроса. Типичная схема большинства чипсетов выглядит так: контроллер шины принимает запрос и ставит его в очередь, которую периодически опрашивает агент транзакций и, извлекая накопившиеся к этому времени запросы, преобразует их в командные пакеты, поступающие на вход планировщика запросов к памяти, который получает запросы сразу от нескольких устройств: процессора, AGP-карты, южного моста и др., стараясь обслуживать всех клиентов максимально эффективно. Спланированные запросы накапливаются в очереди арбитра памяти, который по мере их извлечения распределяет ячейки по физическим адресам, передавая их непосредственно блоку сопряжения с модулями памяти. Теперь вам понятно, чем одни чипсеты отличаются от других?

За словами «контроллер шины» и «контроллер памяти» скрывается целый мир, состоящий из множества узлов и сложно взаимодействующих друг с другом компонентов. Заставить все это хозяйство работать параллельно практически невозможно и потому латентность рядовых чипсетов составляет от десяти до двадцати тактов системной шины! Ума не приложу, как парням из Intel удалось уложиться всего лишь в два. Однако уменьшение латентности еще не увеличивает пропускную способность, и для потоковых приложений (либо же приложений, хранящих обрабатываемые данные в основном в кэше) прирост скорости окажется пренебрежительно мал. По результатам некоторых исследований (сообщения о которых были найдены в Интернете) общий прирост производительности, осуществляемый PAT, составляет всего 2% – 5%. Очень странные цифры! К сожалению, никакой дополнительной информации (что именно и как именно измерялось) авторами исследований не приводится, а потому польза от таких исследований равна нулю. Во всяком случае, в чипсете SiS 655 ничем подобным и не пахнет и, судя по всему, его латентность достаточно велика (к сожалению, из-за отсутствия самого чипсета измерять ее реальное значение не представляется возможным).

Еще одной приятной особенностью чипсета Intel 875P, которую, без сомнения, по достоинству оценят все любители «разгона», является возможность контроля температуры чипов памяти. Не секрет, что даже при работе в штатном режиме память очень сильно нагревается и при плохой циркуляции воздуха внутри корпуса может начать сбоить (кстати, некоторые процессорные вентиляторы этому очень даже способствуют, поскольку направляют поток горячего воздуха прямоком на

модули памяти). Если же рабочую частоту памяти увеличить сверх штатного, то она и вовсе может выйти из строя. Чтобы этого не произошло, чипсет Intel 875P подсчитывает количество обращений к памяти в единицу времени и при достижении некоторого предела начинает вставлять холостые циклы, давая памяти хоть немного остыть. Означает ли это, что память не сможет работать на своей полной пропускной способности (т.е. две порции данных за один такт)? А вот и нет! Последовательное чтение ячеек практически не нагревает микросхему, а вот интенсивное переключение банков – да. Таким образом, увеличив частоту работы памяти сверх штатного, мы сможем увеличить и производительность потоковых приложений, гарантированно оставаясь застрахованными от возможных сбоев, возникающих при хаотичном обращении к различным страницам (что происходит, в частности, при загрузке Windows).

Другой путь – использование внешнего термодатчика, прикрепленного к микросхеме. Чипсет Intel 857P это поддерживает, однако перекладывает заботу об охлаждении памяти на BIOS, что уже не есть хорошо. С другой стороны, дополнительный датчик карман не тянет, и как резервный уровень защиты от перегрева он вполне подойдет.

И наконец, шина CSA (CommunicationsStreaming Architecture) – последняя рассматриваемая нами вкусность чипсета Intel 875P, наличием которой не может похвастаться ни один из его конкурентов. Обеспечивая пропускную способность в 266 Мб/сек, она позволяет значительно разгрузить южный мост от таких прожорливых чудовищ, как, например, Gigabit Ethernet Intel PRO/1000. Весь вопрос в том, сколько производителей интегрируют такой чип на свои материнские платы. Потребность в столь быстрых локальных сетях у общественности еще не возникла и в ближайших перспективах шине CSA суждено оставаться незадействованной.

В остальном же Intel 875P представляется крайне удачным и хорошо сбалансированным чипсетом. Идеальный выбор для тех, кому нужна скорость и у кого есть деньги. Чипсет SiS 655 обеспечивает не сильно худшую скорость, но за значительно меньшую сумму. Однако Intel – бренд (и этим все сказано), а чипсеты сторонних производителей – это все равно, что обезьяна с гранатой. Порой разница между заявленным и действительным столь велика, что начинаешь поневоле задумываться: настолько ли вы богаты, чтобы позволить себе покупать дешевые вещи? В частности, некоторые чипсеты от VIA поддерживают заявленные частоты лишь формально, т.е. они «держат» шину на этой частоте, но внутри самого чипсета данные передаются со скоростью вдвое, а то и вчетверо меньшей. «Маразм», – воскликнете вы. «Нет, просто дешевый чипсет», – возражу я. К слову сказать, в моей книге «Техника оптимизации программ», которая скоро появится в продаже, вы найдете не только развернутое описание архитектуры принципов функционирования оперативной памяти, но и впечатляющий список ошибок реализаций аппаратуры с объяснениями, почему в тех или иных случаях она работает не так, как хотелось бы.

Приложение 1 Выбор и конфигурирование памяти

Поскольку чипсеты Intel 857P и SiS 655 поддерживают далеко не все типы DDR-памяти, то существует ненулевая вероятность приобрести в магазине такой модуль, который на вашем компьютере работать просто не будет, либо же будет работать крайне неэффективно. А потому перед походом в магазин рекомендуем внимательно изучить таблицы, приведенные ниже. Надеемся, они помогут вам сделать правильный выбор (или узнать: подходит ли уже имеющаяся у вас память к новому чипсету или нет).

Что же касается чипсета SiS 655, то в рекламном буклете, выложенном на сайте компании, никакой информации на этот счет нет, а нормальная техническая документация «зажата» и отдается только за деньги. Гм, платить свои деньги, чтобы описывать (читать: продвигать) чипсет компании SiS?! А оно мне нужно?

Таблица 1. Типы памяти, поддерживаемой Intel 875EP.

плотность разрядность	128 Mbit		256 Mbit		516 Mbit	
	x8	x16	x8	x16	x8	x16
тип	SS/DS	SS/DS	SS/DS	SS/DS	SS/DS	SS/DS
емкость	128/286Mb	64 Mb/NA ²	256/512 Mb	128 Mb/NA	512/1024 Mb	256 Mb/NA

Таблица 2. Настройка памяти на максимальную производительность.

частота шины	частота памяти	соотношение частот	пропускная способность памяти	тип памяти	эффективная пропускная способность
4x100 MHz	4x133 MHz	3/4	266 MT/сек	DDR-DRAM	2,1 ГТб/сек
4x133 MHz	4x133 MHz	1/1	266 MT/сек	DDR-DRAM	2,1 ГТб/сек
4x200 MHz	4x133 MHz	3/2	266 MT/сек	DDR-DRAM	2,1 ГТб/сек
4x133 MHz	4x166 MHz	4/5	333 MT/сек	DDR-DRAM	2,7 ГТб/сек
4x200 MHz	4x160 MHz	5/4	320 MT/сек	DDR-DRAM	2,6 ГТб/сек
4x200 MHz	4x200 MHz	1/1	400 MT/сек	DDR-DRAM	3,2 ГТб/сек

Обратите внимание, что DDR 333 выгоднее гонять на 133 MHz шине, но не на 200 MHz!

Таблица 3. Набивка памяти в слоты на Intel 875P и ее влияние на производительность (1 - наивысший рейтинг).

эффективная частота DDR	кол-во модулей	рядов на модуль	режим адресации	режим каналов	P-рейтинг
400 MHz	4	2	динамический	двухканальный	1
400 MHz	2	2	динамический	двухканальный	2
400 MHz	4	1	динамический	двухканальный	2
400 MHz	2	1	динамический	двухканальный	3
400 MHz	4	без разницы	обычный	двухканальный	4
400 MHz	без разницы	2	динамический	одноканальный	5
400 MHz	без разницы	1	динамический	одноканальный	6
400 MHz	без разницы	без разницы	обычный	одноканальный	7

Таблица 4. Набивка памяти в слоты на Intel 875P и ее влияние на производительность (1 - наивысший рейтинг).

эффективная частота DDR	кол-во модулей	рядов на модуль	режим адресации	режим каналов	P-рейтинг
266/333 MHz	2	2	динамический	двухканальный	1
266/333 MHz	4	1	динамический	двухканальный	1
266/333 MHz	2	1	динамический	двухканальный	2
266/333 MHz	4	2	динамический	двухканальный	3
266/333 MHz	без разницы	без разницы	обычный	двухканальный	4
266/333 MHz	2 (1 в каждый канал)	без разницы	динамический	одноканальный	5
266/333 MHz	1	без разницы	динамический	одноканальный	5
266/333 MHz	без разницы	без разницы	динамический	одноканальный	6
266/333 MHz	без разницы	без разницы	обычный	одноканальный	7

Приложение 2 Приблизительная схема взаимодействия памяти и процессора

- Получив запрос на чтение ячейки, процессор выполняет арбитраж и передает чипсету адрес и длину запрошенного блока памяти. При условии, что шина свободна, эта операция укладывается в четыре такта.
- Контроллер шины, получив запрос, ставит его в очередь и, если контроллер памяти свободен, передает ему запрос с началом следующего такта.

- В течение следующего такта контроллер памяти декодирует адрес и ставит его в свою внутреннюю очередь запросов на чтение памяти.
- В следующем такте запрос извлекается из очереди, и контроллер, при необходимости дождавшись прихода фронта тактового импульса микросхемы памяти, передает ей адрес ячейки:
 - I. Если соответствующая страница открыта и банк памяти не находится на регенерации, чипсет выставляет сигнал CAS и передает сокращенный адрес ячейки. Спустя 2-3 такта частоты памяти на шине появляется первая порция считанных данных.
 - II. Контроллер памяти считывает ее за один такт.
 - Синхронный контроллер памяти с началом следующего такта передает считанные данные контроллеру шины и в дальнейшем пересылка осуществляется параллельно с чтением, но с задержкой в один такт.
 - Асинхронный контроллер памяти «благодаря» расхождению частот не может передавать данные одновременно с чтением, и вынужден накапливать их во временном буфере. После завершения пакетного цикла чтения, контроллер памяти по приходу фронта следующего синхромпульса начинает передавать содержимое временного буфера контроллеру шины на требуемой частоте.
 - Примечание: некоторые дешевые чипсеты, в частности VIA KT133/KT266, осуществляют передачу данных внутри чипсета только по фронту импульса, что полностью обесценивает все преимущества шины EV6, на которой работает Athlon, и ее эффективная частота (определяемая, как известно, самым узким местом системы) оказывается равной всего 100/133 MHz.
 - Примечание: если длина запроса превышает длину пакета, то независимо от типа контроллера памяти, данные всегда передаются через временный буфер.
- III. На чтение «хвоста» пакета в зависимости от его длины уходит еще от трех до семи тактов частоты оперативной памяти при использовании памяти типа SDRAM и от двух до трех тактов на DDR соответственно.
- IV. Если длина запроса превышает длину пакета, то мы возвращаемся к пункту I.
- V. Контроллер шины, получив считанные данные, формирует запрос на передачу данных от чипсета к процессору и ставит его в очередь, на что расходуется один такт.
- VI. Если в очереди не находится ничего другого, и шина никем не занята, контроллер шины извлекает запрос из очереди и «выстреливает» его в шину, передавая за один такт одну, две или четыре порции данных (на K6/P-II/P-III, Athlon и P-4 соответственно).
- VII. Как только запрошенная ячейка попадает в процессор, она становится немедленно доступной для обращения, даже если пакетный цикл передачи еще не завершен.

VIII. Все! Остается лишь добавить латентность кэш-контроллеров всех иерархий и латентность самого процессора, но это уже тема другого разговора, к оперативной памяти прямого отношения не имеющая.

- если требуемая DRAM-страница закрыта, но банк не находится на регенерации, контроллер памяти передает адрес строки, вырабатывает сигнал RAS, ждет 2 или 3 такта, пока микросхема его «переварит», и переходит к сценарию I.
- если же банк находится на регенерации, контроллеру приходится подождать от одного до трех тактов, пока она не будет завершена.

Приложение 3 Латентность против пропускной способности

С точки зрения пользователя PC главная характеристика памяти – это скорость или, другими словами, ее быстродействие. Казалось бы, измерить быстродействие просто. Достаточно подсчитать количество информации, выдаваемой памятью в единицу времени (скажем, мегабайт в секунду), и... ничего не получится! Дело в том, что время доступа к памяти непостоянно и в зависимости от характера обращений варьируется в очень широких пределах. Наибольшая скорость достигается при последовательном чтении, а наименьшая – при чтении вразброс. Но и это еще не все! Современные модули памяти имеют несколько независимых банков и потому позволяют обрабатывать более одного запроса параллельно.

Если запросы следуют друг за другом непрерывным потоком, непрерывно генерируются и ответы. Несмотря на то, что задержка между поступлением запроса и выдачей соответствующего ему ответа может быть весьма велика, в данном случае это не играет никакой роли, поскольку латентность (т. е. величина данной задержки) полностью маскируется конвейеризацией и производительность памяти определяется исключительно ее пропускной способностью. Можно провести следующую аналогию: пусть сборка одного отдельного взятого Мерседеса занимает, скажем, целый месяц. Однако если множество машин собирается па-

раллельно, завод может выдавать хоть по сотне Мерседесов в день, и его «пропускная способность» в большей степени определяется именно количеством сборочных линий, а не временем сборки каждой машины. В настоящее время практически все производители оперативной памяти маркируют свою продукцию именно в пропускной способности, но наблюдающийся в последнее время стремительный рост пропускной способности адекватного увеличения производительности приложений, как это ни странно, не вызывает. Почему?

Основной камень преткновения – фундаментальная проблема зависимости по данным. Рассмотрим следующую ситуацию. Пусть ячейка N 1 хранит указатель на ячейку N 2, содержащую обрабатываемые данные. До тех пор пока мы не получим содержимое ячейки N 1, мы не сможем послать запрос на чтение ячейки N 2, поскольку еще не знаем ее адреса. Следовательно, производительность памяти в данном случае будет определяться не пропускной способностью, а латентностью. Причем не латентностью микросхемы памяти, а латентностью всей подсистемы памяти – кэш-контроллером, системной шиной, набором системной логики... Латентность всего этого хозяйства очень велика и составляет порядка 20 тактов системной шины, что многократно превышает полное время доступа к ячейке оперативной памяти. Таким образом, при обработке зависимых данных быстродействие памяти вообще не играет никакой роли: и SDRAM PC100, и RDRAM-800 покажут практически идентичный результат!

Причем описываемый случай отнюдь не является надуманным, скорее наоборот, это типичная ситуация. Основные структуры данных (такие, как деревья и списки) имеют ярко выраженную зависимость по данным, поскольку объединяют свои элементы именно посредством указателей, что «съедает» весь выигрыш от быстродействия микросхем памяти.

¹ На самом деле кэш-линейки P-4 делятся на две половинки, загружающих свое содержимое независимо друг от друга.

² NA: not available – не поддерживается.

