

## ОСОБЕННОСТИ RISC-ПРОЦЕССОРОВ

Наиболее сложновывполнимое из упомянутых требований — первое, в соответствии с которым каждая команда должна выполняться не более чем за один такт. Однако несмотря на существенное упрощение, необходимо, чтобы устройство управления по-прежнему выполняло операции по чтению кода команды, декодированию, подготовке операндов, собственно выполнению команды и перемещению результата в соответствующий регистр. Если учесть, что каждая из этих операций обязана синхронизироваться с фронтом (или спадом) сигнала тактового генератора, то единственно возможный вариант решения проблемы — использование так называемого конвейера команд. При этом по первому перепаду сигнала тактового генератора в соответствующем устройстве выделяется код операции, который затем передается в устройство декодирования; по второму — в устройстве извлечения выделяется код операции следующей команды, а в устройстве декодирования происходит декодирование первой команды и передача ее в соответствующее исполнительное устройство — АЛУ, сопроцессор и т. д. Третий перепад сопровождается извлечением кода операции третьей команды, декодированием второй и подготовкой данных для выполнения первой.

Таким образом, по каждому из тактовых импульсов на конвейер вступает новая команда, и несколько уже обрабатываются на разных его ступенях. Одновременно, также по каждому из тактовых импульсов, его покидает одна выполненная команда. И хотя на выполнение каждой затрачивается по-прежнему от четырех до семи-восьми тактов (в рассмотренном выше условном конвейере — пять), каждый из них сопровождается, как это и требовалось, выполнением одной команды. Следовательно, если для CISC-процессоров конвейер команд является обязательным (хотя и желательным) элементом, то для RISC-процессоров он обязателен. Отметим, что большинство RISC-процессоров имеют не один, а несколько (от двух до четырех) конвейеров, за что они получили название суперскалярных (в отличие от скалярных — одноконвейерных).

Следующая особенность RISC-процессоров — высокая степень дробления конвейера. В рассмотренном выше примере он состоит из пяти ступеней: извлечения кода операции, декодирования, подготовки операндов, исполнения, сохранения результата. Реально RISC-процессоры характеризуются семи-десятиступенчатыми конвейерами. С увеличением числа ступеней действия, выполняемые на каждой из них, все более и более упрощаются. Последнее означает, что уменьшается число необходимых для этого логических элементов и появляется возможность повысить тактовую частоту процессора. Не случайно RISC-процессоры преодолели барьер 100 МГц существенно раньше CISC-процессоров. К осени 1997 г. самыми высокочастотными из последних были 266-мегагерцевые Pentium II, в то время как RISC-процессоры давно превзошли рубеж 300 МГц, а в

1998 г. ожидается анонсирование фирмой DEC новинки с тактовой частотой 500 МГц.

Третья особенность — использование большого числа регистров. Как правило, в RISC-процессорах их не менее 32. Особенно выделяются изделия фирмы MIPS, имеющие сложную многооконную регистровую структуру и позволяющие адресовать до 136 регистров. Подобная свобода (семейство x86 имеет всего восемь регистров общего назначения) позволяет снизить число обращений к относительно медленной оперативной памяти в полтора-два раза (в сравнении с CISC-процессорами), что опять-таки положительно сказывается на росте реальной производительности системы. Добавим к этому, что все RISC-процессоры содержат системы управления кэш-памятью второго уровня, позволяющие работать с ней на максимальной скорости (в CISC-изделиях взаимодействие с кэш-памятью обычно происходит на частоте, более низкой, чем частота самого процессора).

Еще одна особенность RISC-процессоров — развитые средства прогнозирования ветвлений и переходов. В программе для семейства x86 команда перехода встречается в среднем через каждые шесть, в программах для RISC-процессоров, команды которых проще, — через каждые 10...12 команд. Встретив команду условного перехода, процессор должен сделать предположение о том, выполнится условие или нет, и в зависимости от этого начать предвыборку команд либо с адреса предполагаемого перехода, либо с адреса, следующего за текущим.

Если переход предсказан неправильно, процессору необходимо удалить со всех ступеней конвейера команды, относящиеся к неверно сделанному предположению, и перезагрузить конвейер. Это особенно сказывается на работе суперскалярных процессоров — на разных ступенях трех-четырех конвейеров может находиться довольно много команд. Их удаление с последующей перезагрузкой приводит к тому, что в течение нескольких тактов конвейер не покидает ни одна команда. Процессор, в котором это случается часто, теряет 20...30 % своей производительности. Поэтому RISC-процессоры характеризуются весьма эффективными механизмами предсказания ветвлений. Кроме того, они содержат устройства, позволяющие выбрать те из команд в предсказанном переходе, которые можно выполнить прежде, чем станет ясно, правильно ли был предсказан переход.

### RISC-ПРОЦЕССОР PowerPC 601

Познакомимся поближе с конкретными представителями группы RISC-процессоров. Начнем с изделий, которые многими рассматриваются как реальная альтернатива процессорам семейства x86. Это — продукция альянса Apple-IBM-Motorola, объединенная названием PowerPC.

Осенью 1991 г. компании Apple, IBM и Motorola сообщили о решении создать семейство RISC-процессоров широкого спектра назначения — от серверов верхнего уровня и рабочих станций до на-

стоящих и переносных компьютеров. За основу проекта был взят процессор Power (Performance Optimized With Enhanced RISC) фирмы IBM, разработанный ею для своих широко известных рабочих станций RS/6000. Проект был рассчитан на 10 лет, однако первые процессоры должны были появиться уже через два года после подписания соглашения. И на осенней выставке Comdex'93 новые изделия были продемонстрированы: процессор, получивший название PowerPC 601, характеризовался 32-рядной внутренней структурой, содержал на кристалле площадью 121 мм<sup>2</sup> 2,8 млн транзисторов и работал на частотах 50, 66 и 80 МГц.

PowerPC 601 относится к суперскалярным структурам — за один такт может выполнять до трех команд. Он оперирует с 8-, 16- и 32-битными целыми данными, а также с 32- и 64-битными числами с плавающей запятой. Его производительность на целочисленных тестах примерно соответствует производительности процессора Pentium с той же тактовой частотой, а на тестах с плавающей запятой — примерно на треть выше.

На кристалле PowerPC 601 располагаются три исполнительных устройства: целочисленное (IU), для операций с плавающей запятой (FPU) и для обработки переходов (BPU). Работают они параллельно, в значительной мере независимо друг от друга, благодаря чему процессор и может во многих случаях (но не всегда) выполнять три команды за один такт.

Помимо названных устройств, на кристалле располагаются кэш-память команд и данных объемом 32 Кбайт и устройство управления памятью (MMU). Процессор имеет 64-битную шину данных и 32-битную адресную шину. При работе с памятью он поддерживает как по-тактовую, так и пакетную передачу данных. Последняя характеризуется тем, что адресная информация устанавливается на выводах процессора и считывается только в начале передачи. После этого передаются 16 байт, располагающиеся в памяти непосредственно друг за другом. Передача этого пакета происходит довольно быстро, так как устройства управления памятью не тратят время на лишние циклы обработки выведенной адресной информации.

Отметим, что устройство обмена с памятью имеет отдельные буферы для чтения и для записи. Это позволяет процессору не тормозить свою работу, если, например, в настоящий момент запись в память по каким-то причинам невозможна — процессор заносит данные в буфер и продолжает выполнять программу.

Команды из кэш-памяти поступают в устройство подготовки команд, содержащее очередь из восьми команд и BPU. Устройство подготовки определяет адрес следующей считываемой команды, управляет опережающим считыванием команд и блокировкой конвейера. BPU просматривает нижнюю половину очереди и предсказывает вероятное расположение команды, на которую будет произведен переход. Выполнение команд за предполагается точкой условного перехода начинается до момента точного предсказания адреса перехода, но не доводится до конца. Если эти команды