

Тема 13

УПРАВЛЕНИЕ ПАМЯТЬЮ ЭВМ, РАСШИРЕНИЕ АДРЕСНОГО ПРОСТРАНСТВА, ДИНАМИЧЕСКОЕ РАСПРЕДЕЛЕНИЕ ПАМЯТИ

13.1. Проблемы управления памятью и расширения адресного пространства

Оперативное запоминающее устройство ЭВМ, как и ее центральный процессор, являются наиболее важными элементами ЭВМ, определяющими эффективность работы ЭВМ в целом. Важность памяти ЭВМ определяется тем, что *никакая программа не может быть выполнена компьютером, если она не находится в его оперативной памяти.*

В связи с этим сложность и время выполнения решаемых компьютером все более сложных задач и увеличивающихся в размерах программ все в большей степени определяются объемом его оперативной памяти.

Проблема использования физического объема памяти компьютера еще больше усугубляется широко используемой организацией работы ЭВМ в *многозадачном (мультипрограммном) и многопользовательском* режимах. В мультипрограммных системах размещение всех исполняемых программ полностью в оперативной памяти в большинстве случаев просто невыполнимо, так как современные программы могут иметь большие и очень большие размеры, в то время как объем представленной в ЭВМ оперативной памяти всегда ограничен.

Однако на самом деле нет принципиальной необходимости в том, чтобы все части программы одновременно находились в оперативной памяти, так как в конкретный момент времени

в работе программы участвует определенная сравнительно небольшая часть ее кода. Это позволяет размещать в оперативной памяти только части программы, используемых именно в данный период времени, а неиспользуемые ее части (блоки, модули) могут располагаться на внешнем запоминающем устройстве до того момента, когда возникнет в них потребность. Также следует учитывать тот факт, что, разрабатывая свою программу, пользователь, как правило, даже не знает, в комбинации с какими другими программами будет выполняться его программа, и какое место в оперативной памяти отведет ей операционная система, которая должна, в частности, обеспечивать возможность *независимой работы* программистов над созданием своих программ, подлежащих использованию в мультипрограммном режиме.

В связи с этим, помимо стремления к увеличению размеров доступного адресного пространства оперативной памяти ЭВМ, большое значение приобретает проблема эффективного управления этим адресным пространством, его динамического распределения между различными задачами (программами) в процессе их выполнения.

Помимо возможности создания и размещения в ЭВМ какого-то количества физической запоминающих ячеек, наиболее важным фактором, определяющим объем адресного пространства оперативной памяти ЭВМ, является количество разрядов в адресах, с которыми может оперировать ее центральный процессор. Так в шестнадцатиразрядных ЭВМ пользователь может непосредственно использовать в программе максимум $2^{16} = 65\,536$ адресов, выражаемых шестнадцатиразрядными двоичными числами.

Важно понимать, что для того чтобы ЭВМ могла обеспечить обращение к большому адресному пространству, в ней обязательно должны быть аппаратные средства, позволяющие формировать для обращения к оперативной памяти адрес с большим количеством разрядов. Оказывается, что такие возможности могут быть реализованы и в большинстве современных шестнадцатиразрядных ЭВМ.

Процессоры, оперирующие словами с большим количеством разрядов, например 32- и 64-разрядные, позволяют непосредственно адресоваться к значительно большому адресному пространству. Однако и для них, может быть даже в большей степени, остается важной проблема эффективного динамического

распределения имеющейся памяти между большим числом задач и пользователей.

Ниже будут рассмотрены основные архитектурные решения, используемые для увеличения доступного адресного пространства шестнадцатиразрядных ЭВМ и управления такой расширенной памятью. При этом, однако, надо иметь в виду, что эта проблема решается комплексом аппаратно-программных средств, то есть не только определенными архитектурными особенностями построения процессора ЭВМ, но и тесно связанными с ними программными средствами операционной системы ЭВМ.

13.2. Физическое и виртуальное адресные пространства

При решении проблемы расширения адресного пространства и управления памятью ЭВМ принципиально важной является концепция разделения *так называемых физического и виртуального* адресных пространств.

Под *физической памятью* ЭВМ понимают множество имеющихся в аппаратуре ЭВМ *ячеек оперативной памяти*.

Виртуальной памятью называют адресное пространство, с которым оперирует пользователь-программист в своих программах.

При этом разделение таких двух видов памяти — физической и виртуальной — подразумевает, что в общем случае используемые конкретной программой физические и виртуальные адреса не совпадают.

Концепция использования понятия *виртуальная память* предполагает, что пользователь при подготовке своей программы имеет дело не с физическим набором реальных запоминающих ячеек оперативной памяти, имеющей место в составе конкретной вычислительной системы, а с абстрактной виртуальной (то есть физически не существующей) памятью.

Размер виртуальной памяти, которым оперирует в этом случае программист, равен всему адресному пространству, определяемому размером адресных полей в форматах команд и базовых регистров. То есть программист как бы имеет в своем распоряжении все адресное пространство системы независимо от объема ее фактической физической памяти и независимо от размеров и расположения областей памяти, необходимых

для работы других программ, участвующих в мультипрограммной обработке.

На всех этапах подготовки программ, включая загрузку в оперативную память, программа представляется в *виртуальных адресах*, начинающихся от нулевого адреса. Автор программы заранее не знает, где в физической памяти будут располагаться модули его программы. Преобразование виртуальных адресов в реальные физические адреса действующей памяти производится во время самого исполнения машинной команды и выполняется с помощью специальных аппаратных средств компьютера и системных программных средств.

Эти средства осуществляют отображение виртуальных адресов одной или нескольких программ на конкретные адреса физического адресного пространства запоминающего устройства компьютера. При этом заметим, что отображение это может быть более или менее эффективным с точки зрения общих критериев решения компьютером пользовательских задач.

Преобразование виртуальных адресов в физические упрощается, если физическую и виртуальную память разбить на блоки, содержащие одинаковое число байтов, и осуществлять поблочное отображение блоков виртуальной памяти в блоки памяти физической. Такие блоки обычно называются *страницами*, или *сегментами*. Страницам виртуальной и физической памяти присваивают номера, называемые номерами соответствующих виртуальных и физических страниц. Каждая физическая страница способна хранить одну из виртуальных страниц. Порядок расположения (нумерация байтов) в виртуальной и физической страницах сохраняется одним и тем же.

В мультипрограммной системе страничная организация памяти дает определенные преимущества. Когда новая программа загружается в оперативную память, она может быть направлена в любые свободные в данный момент физические страницы независимо от того, расположены они подряд или нет. Не требуется перемещения информации в остальной части памяти. Страничная организация позволяет сократить объем передачи информации между внешним запоминающим устройством и оперативной памятью. Программа может загружаться в оперативную память не целиком. Вначале в оперативную память загружается начальная страница программы, которой передается управление. Другие страницы программы не должны загружаться до тех пор, пока они действительно

не понадобятся. Если по ходу программы делается попытка выборки слов из другой страницы, то производится автоматическое обращение к операционной системе, которая осуществляет загрузку требуемой страницы.

13.3. Расширение адресуемого пространства в шестнадцатиразрядных ЭВМ

Рассмотрим основные принципы архитектурных решений, используемых для управления памятью в 16-разрядных процессорах ЭВМ. В этих ЭВМ физическое адресное пространство может быть больше виртуального, ограниченного разрядностью виртуального адреса и составляющего 2^{16} байт.

Модификация, то есть преобразование используемого в программе виртуального адреса в физический адрес расширенной памяти, осуществляется в ЭВМ путем прибавления к виртуальному адресу так называемой константы перемещения, смещающей виртуальный адрес в соответствующее место физической памяти, как это показано на рис. 13.1.

		Виртуальный адрес																						
		0	1	1	0	1	0	1	1	1	0	0	1	0	1	1	0							
		15					8			7	6							0						
+	Константа перемещения																							
		0	1	0	0	1	0	1	1	1	0	0	1	1	0	1	0	0	0	0	0			
=		Физический адрес																						
		0	1	0	0	1	1	0	1	0	1	1	1	0	1	1	0	1	0	1	1	0		
		21				15																		0

Рис. 13.1. Преобразование виртуального адреса в физический

Для формирования константы перемещения, определяющей фактический адрес в физической памяти, в состав процессора вводят один или несколько дополнительных регистров, в которые константы перемещения устанавливаются программным путем. Понятно, что, вообще говоря, число разрядов в константе перемещения должно быть такое же, как и в адресе физической памяти (в примере на рис. 13.1 это 22 разряда). Однако также очевидно, что в шестнадцатиразрядной ЭВМ числа с таким количеством разрядов мы формировать и размещать программным путем не можем. Поэтому для того, чтобы обеспечить возможность формирования констант переме-

щения программным путем, регистры для хранения констант перемещения делают также шестнадцатиразрядными. А чтобы из шестнадцатиразрядного числа получить двадцатидвухразрядное, содержащееся в регистре шестнадцатиразрядное число аппаратным образом дополняют справа шестью нулями.

То есть, при формировании физического двадцатидвухразрядного адреса в нашем усовершенствованном процессоре к исходному виртуальному адресу прибавляется содержимое регистра, содержащего константу перемещения дополненную справа шестью нулями как это показано на рис. 13.1.

Вследствие того, что эти дополняемые младшие разряды константы перемещения не могут изменяться, точность, с которой она может позиционировать виртуальный адрес в физической памяти, определяется числом таких не модифицируемых младших разрядов (на рис. 13.1 это шесть разрядов). Другими словами, страница виртуальной памяти отображается не в любое место физической памяти, а дискретно, по адресам, кратным числу, определяемому количеством добавляемых к содержимому регистра с константой перемещения не модифицируемых младших разрядов. Сама записываемая в такой регистр константа перемещения задает фактически смещение адреса начала физической страницы (сегмента) от начала физической памяти, а виртуальный адрес задает относительный адрес внутри физической страницы (сегмента).

В связи с этим регистры процессора, используемые для указания констант перемещения, иногда называются регистрами адреса страниц (РАС).

В процессорах Intel они называются сегментными регистрами. Например, в процессорах Intel 8086, 80286 имеется четыре шестнадцатиразрядных сегментных регистра (CS, DS, SS и ES), которые используются для различных целей. Регистр CS указывает, из какого сегмента физической памяти осуществляется выборка команд, регистр DS указывает, из какого сегмента выбираются данные (операнды), регистр SS идентифицирует текущий сегмент стека и, наконец, регистр ES — это текущий дополнительный сегмент данных. Организация выполнения в ЭВМ нескольких пользовательских программ, размещенных в разных областях физической памяти, состоит в программировании сегментных регистров таким образом, чтобы виртуальные адреса каждой пользовательской программы переадресовывались в соответствующие, отведенные для них области физической памяти.

13.4. Страничная организация памяти

Недостатком использования для преобразования виртуальных адресов в физические единственного регистра смещения является то, что в этом случае виртуальные адреса пользовательской программы преобразуются в физические адреса единым массивом с использованием общей константы смещения. Это затрудняет оптимальное динамическое распределение пространства физической памяти между пользовательскими программами.

Более интересным является техническое решение использования для задания смещения не единственного регистра, а набора регистров (они называются регистрами адреса страниц), позволяющее гораздо более эффективно организовать отображение страниц виртуального адресного пространства пользовательских программ в физическую память компьютера.

Использование набора регистров для задания смещений при преобразования виртуальных адресов иллюстрируется рис. 13.2.

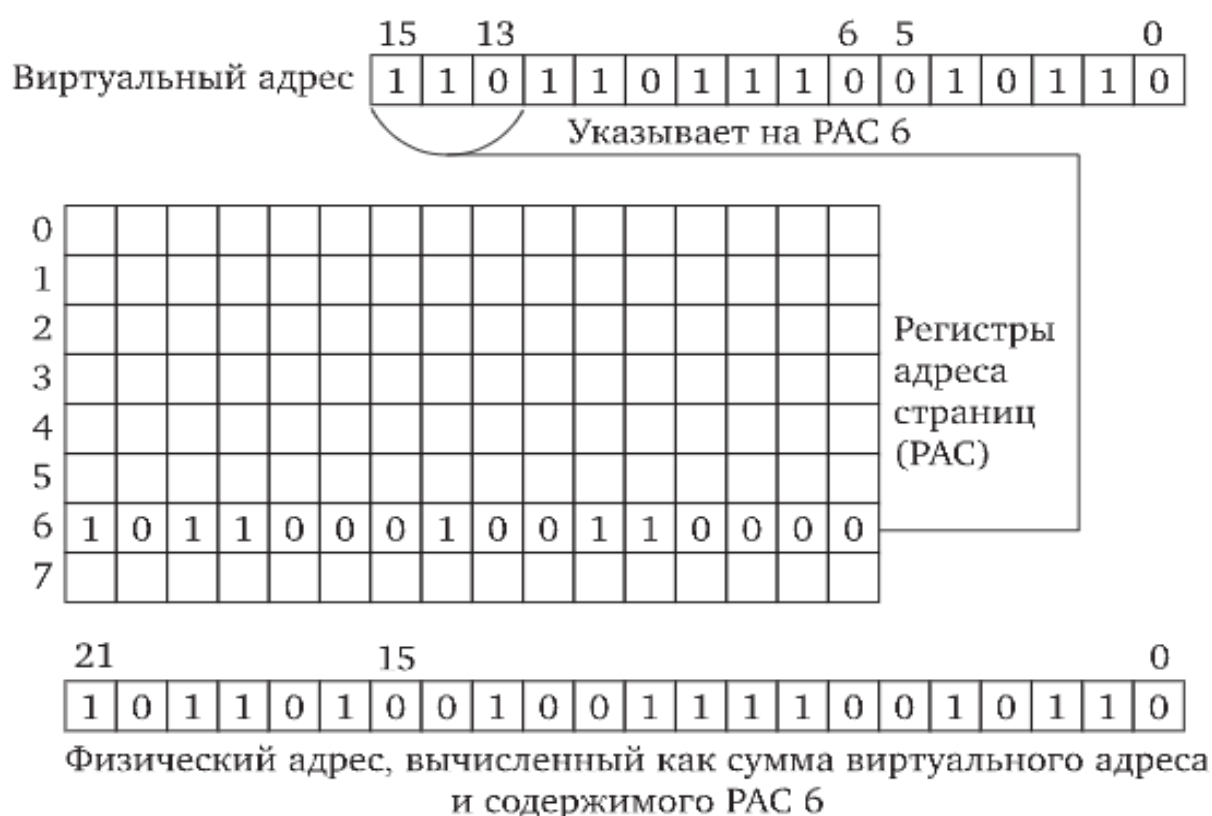


Рис. 13.2. Преобразование виртуальных адресов в физические с использованием регистров адреса страниц

Как видно из рис. 13.2, в данном случае для указания констант перемещения используется не один, а восемь регистров

адреса страниц. Для преобразования конкретного виртуального адреса пользовательской программы в физический используется конкретная константа смещения, извлекаемая из одного из этих регистров. При этом номер регистра адреса страницы, который используется для переадресации конкретного виртуального адреса, определяется тремя старшими разрядами данного виртуального адреса.

Такой способ формирования физических адресов позволяет отображать страницы непрерывного виртуального адресного пространства, следующие последовательно друг за другом, в восемь страниц физической памяти произвольным образом, размещая их в памяти независимо друг от друга (рис. 13.3).

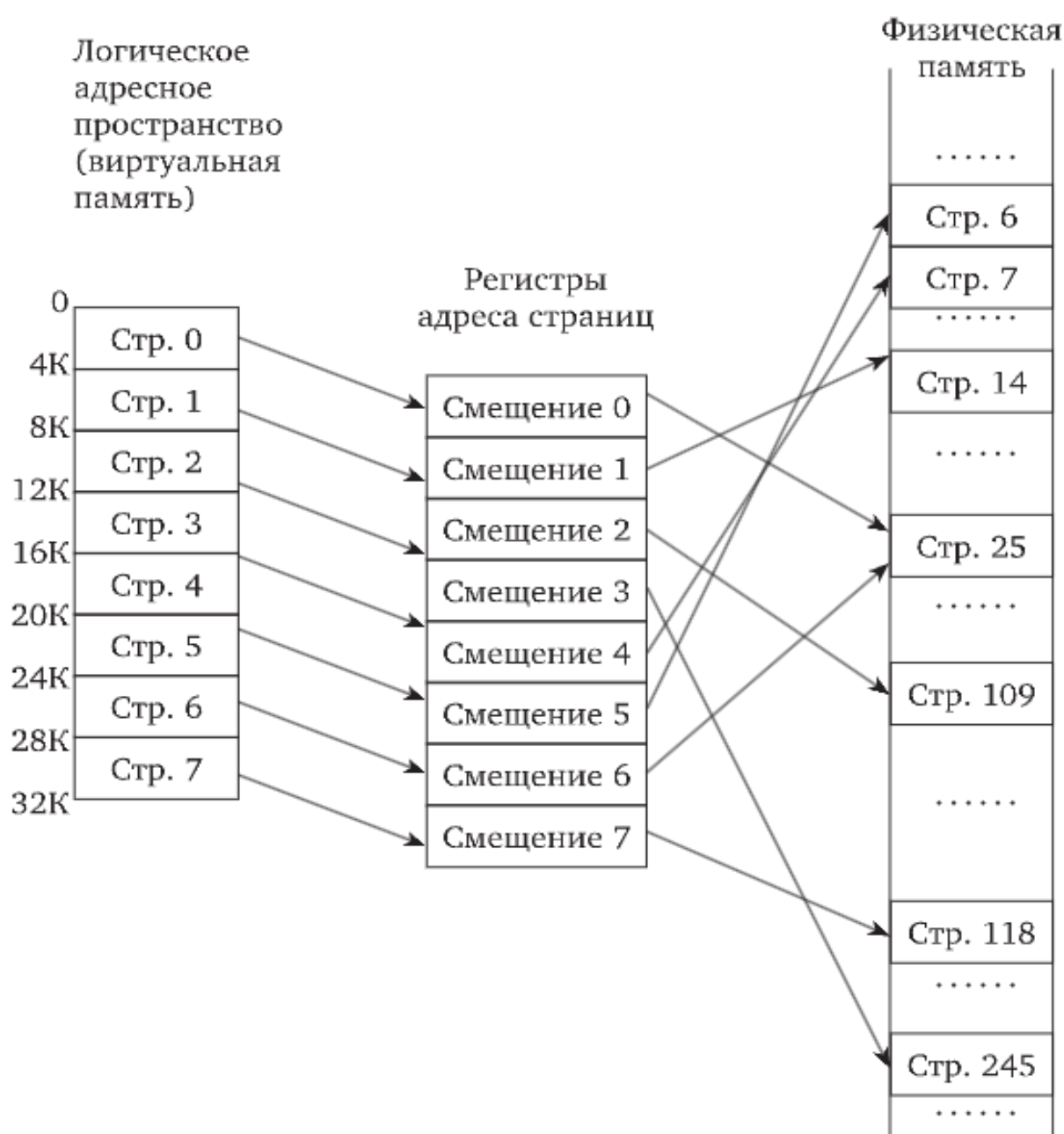


Рис. 13.3. Отображение виртуальных страниц в физические

Содержащиеся в регистрах адреса страниц восемь констант перемещения, определяющих конкретное размещение фрагментов пользовательской программы в физической памяти ЭВМ, иногда называются картой памяти или таблицей страниц программы.

Для каждой пользовательской программы используется своя карта памяти, обеспечивающая отображение адресов страниц именно этой программы в свои области физической памяти. При необходимости отдельные страницы разных программ могут даже перекрываться. Установка начальных значений регистров адреса страниц (карты памяти) и их изменение в процессе работы осуществляется программным путем и обычно выполняется программными средствами операционной системы, обеспечивающими распределение пространства оперативной (физической) памяти между программами.

Аппаратно-программный комплекс, включающий в себя средства в структуре процессора для преобразования используемых в командах виртуальных адресов и системные программные средства, предназначенные для распределения между задачами физического адресного пространства, иногда называют *диспетчером памяти*.

Как правило, аппаратная часть диспетчера памяти процессора включает в себя не только *набор регистров адреса страниц*. Обычно в процессоре в паре с каждым из этих регистров используется еще так называемый *регистр описания страницы*. Регистры описания страницы используются для отражения информации, связанной с характером работы с конкретной страницей. Например, установка режима доступа к данной странице — запись/чтение/запрет доступа, время последнего доступа к странице для контроля того, осуществлялось ли и когда обращение к данной странице и др.

13.5. Управление памятью в многозадачном режиме

В тридцатидвух- и шестидесятичетырехразрядных ЭВМ используемые в командах виртуальные адреса имеют соответственно по 32 и 64 разряда. Поэтому проблема расширения адресуемого пространства памяти для них не является актуальной. Однако проблема управления памятью с целью эффективного распределения ее адресного пространства между

различными процессами становится даже более актуальной. Многозадачный режим работы является для таких компьютеров основным.

Как уже говорилось выше, основная идея управления адресным пространством компьютера и его оптимальным распределением между различными процессами и пользовательскими задачами основывается на концепции разделения виртуального и физического адресных пространств и автоматического преобразования используемых в командах виртуальных адресов для их отображения в физическую память. В отличие от шестнадцатиразрядных ЭВМ, пользователю тридцатидвух- и шестидесятичетырехразрядных компьютеров при разработке программ предоставляется адресное пространство начиная с нулевого адреса и до максимального, равного соответственно $2^{32} - 1$ и $2^{64} - 1$. Разбиение программного кода на страницы и использование постраничного преобразования адресов виртуальных страниц в физические страницы позволяет выполнять компьютеру те или иные задачи, загружая в физическую память не все страницы соответствующей программы, а только ту или те, которые необходимы в данный момент времени, сохраняя остальные на внешнем запоминающем устройстве и подгружая их в память по мере возникновения необходимости.

Рассмотрим для примера управление памятью для тридцатидвухразрядной ЭВМ. Виртуальное адресное пространство размером 2^{32} адресов разделяется на страницы, например по 512 байт (9 разрядов в адресе). Программа строится в виде последовательности набора страниц, пронумерованных от нуля до некоторого значения. Старшие разряды (с 9-го по 31-й) виртуального адреса определяют номер виртуальной страницы.

Физическая память также разделяется на страницы по 512 байт. Адресация страниц физической памяти выполняется с помощью 23-разрядного адреса ($23 = 32 - 9$). Младшие 9 (с 0-го по 8-й) разрядов физического адреса указывают номер байта внутри страницы, а старшие 23 разряда — номер страницы в физической памяти.

Отображение виртуального адресного пространства в физическую память осуществляется аппаратными средствами процессора и системным программным обеспечением — диспетчером памяти, которые, кроме преобразования виртуальных адресов в физические, обеспечивают распределение памяти

между исполняемыми процессами и управление доступом к памяти, то есть ее защиту.

Общая структура отображения виртуальных адресов в физическую память представлена на рис. 13.4.

На рис. 13.4 показано соответствие между виртуальными и физическими страницами, устанавливаемое страничными таблицами (таблицами преобразования адресов) для двух программ А и В. В общем случае количество виртуальных страниц этих программ может превышать размеры оперативной памяти компьютера.

Как видно из рис. 13.4, физические страницы, в которые должны отображаться виртуальные страницы программ, в текущий момент времени могут содержаться как в оперативной, так и во внешней памяти компьютера. Для этих целей во внешней памяти отводится специальная область.

Такой способ страничной организации адресного пространства оперативной физической памяти ЭВМ позволяет размещать непосредственно в оперативной памяти только часть страниц процесса, а другую часть его страниц, которые в данный момент процессором не используются, размещать уже в соответствующей области на внешнем запоминающем устройстве (на диске). Эта область дискового пространства как бы имитирует недостающую для размещения этого процесса оперативную физическую память. Страницы процесса, размещенные на внешнем запоминающем устройстве, загружаются с него в оперативную память по мере необходимости обращения процессора к расположенным в этих страницах командам. При отсутствии в оперативной памяти свободного пространства эти страницы замещают в оперативной памяти те страницы, которые в данный момент не используются. Эти неиспользуемые страницы должны быть выгружены на диск, освобождая место для загрузки нужных активных страниц с диска. Такая процедура выгрузки не активных страниц на диск и загрузки с диска на их место нужных страниц называется *обменом страниц*, или *свопингом* (от англ. *swap*).

Область памяти внешнего запоминающего устройства, отведенную для размещения внешней части физической памяти, иногда называют файлом свопинга, файлом подкачки, а иногда и виртуальной памятью. Хотя, как следует из изложенного выше материала, мы под виртуальной памятью понимаем другое.

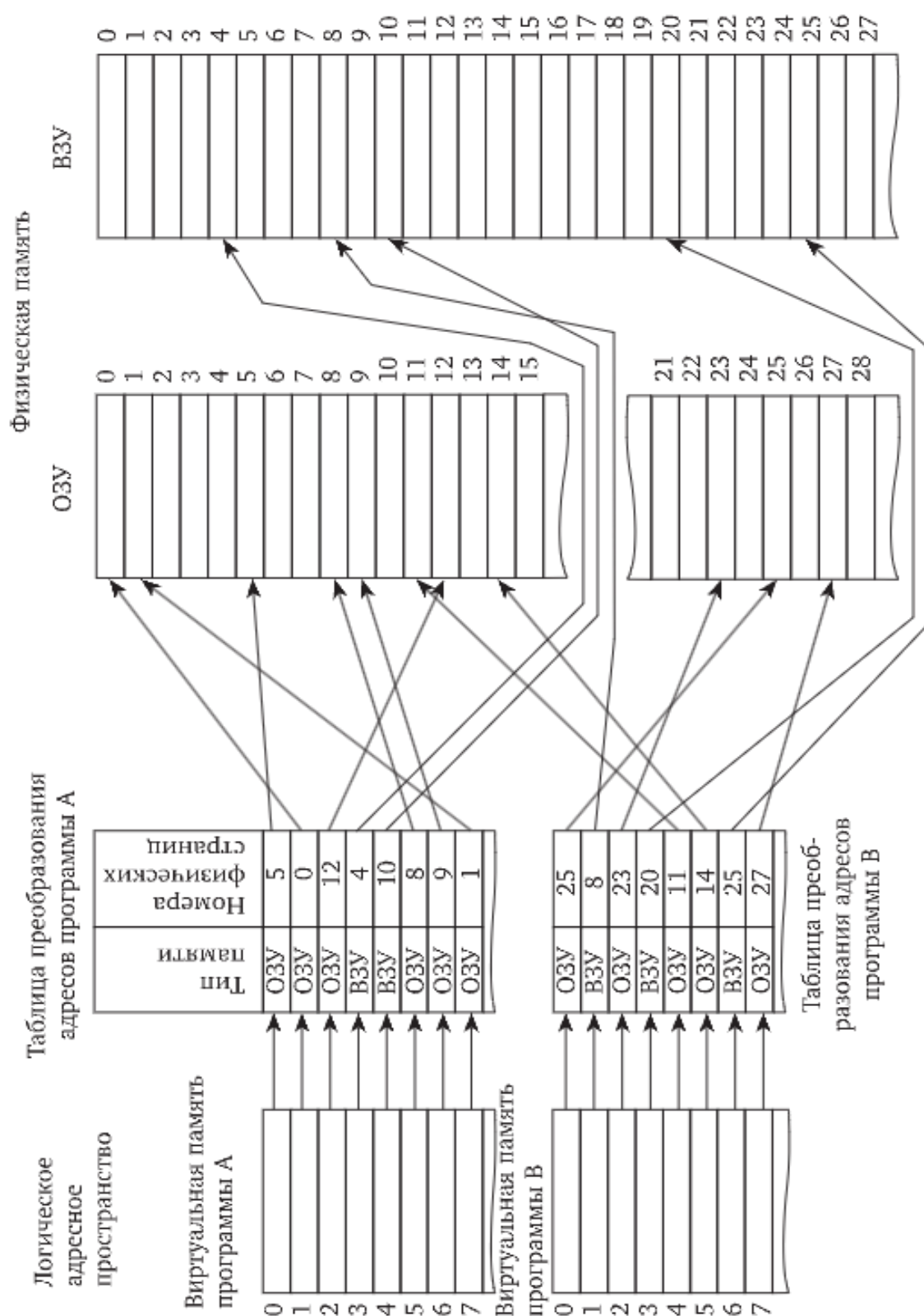


Рис. 13.4. Управление памятью ЭВМ

Должно быть понятно, что для того, чтобы система могла принимать решение о том, какие страницы могут быть выгружены на диск для освобождения места в оперативной памяти для вновь загружаемых страниц, система должна отслеживать и хранить дополнительную информацию о каждой странице, например, было или нет обращение к странице, время последнего обращения к ней. Возможно также отслеживать и хранить количество обращений к странице.

Очевидно, что выполняемые в компьютере действия по обмену страницами физической памяти между оперативной и внешней памятью компьютера требуют определенных, и немалых, временных затрат, иногда существенно замедляя выполнение программ, которым принадлежат обмениваемые страницы. Отсюда вытекает вывод о важности увеличения размера оперативной памяти для повышения скорости работы выполняемых компьютером многостраничных программ. При прочих равных условиях увеличение размера адресуемой оперативной памяти снижает частоту операций по обмену страниц между ней и внешней памятью, тем самым снижая временные затраты на эти операции. И напротив, при недостаточном размере оперативной памяти возрастающая частота обмена страниц может привести к совершенно неудовлетворительной скорости работы компьютера.

При выполнении программы диспетчер памяти должен каждый виртуальный адрес, генерируемый процессом, преобразовать (транслировать) в соответствующий физический адрес.

Для трансляции виртуальных адресов в физические используется структура данных, называемая *таблицей страниц* (*таблицей преобразования адресов*). Эта таблица содержит информацию, необходимую для трансляции адресов. Таблица страниц представляет собой набор записей, по одной на каждую страницу виртуальной памяти. Каждая запись указывает на *режим доступа* процессора (чтение, запись, недоступна) к соответствующей странице физической памяти, *расположение страницы* в физической памяти или на диске, *номер страницы* в физической памяти, если она там находится.

Страничная таблица для каждой программы формируется операционной системой в процессе распределения памяти и корректируется ею каждый раз, когда в распределении памяти происходят изменения. Процедура обращения к памяти состоит в том, что номер виртуальной страницы извлекается

из ее адреса (рис. 13.5) и используется для обращения к страничной таблице для определения номера соответствующей физической страницы.



Рис. 13.5. Формирование физического адреса

Этот номер вместе с номером байта, взятым непосредственно из виртуального адреса, представляет собой физический адрес, по которому происходит обращение к оперативной памяти. Если страничная таблица указывает на размещение требуемой страницы во внешней памяти, то обращение к оперативной памяти не может состояться немедленно. Операционная система должна организовать передачу нужной страницы из внешней памяти в оперативную.

Каждая из программ, обрабатываемых в мультипрограммном режиме, обладает своей виртуальной памятью, и для каждой программы создается своя страничная таблица (таблица преобразования адресов). При этом все программы делят между собой одну общую физическую память, размещаемую частично в оперативной и частично во внешней памяти.

Страничные таблицы программ хранятся в системной области оперативной памяти компьютера. Обращение к нужной строке активной страничной таблицы в оперативной памяти происходит по адресу, который определяется номером активной программы и номером виртуальной страницы (см. рис. 13.4).

Для ускорения преобразования адресов в компьютере может использоваться небольшая сверхоперативная память, в которую передается из оперативной памяти страничная таблица активной программы.

Контрольные вопросы

1. В чем состоит проблема управления памятью ЭВМ, в чем отличие этой проблемы для шестнадцати-, тридцатидвух- и шестидесятичетырехразрядных ЭВМ?
2. В чем состоит концепция разделения понятий физического адресного пространства (физической памяти) и логического (логической, виртуальной памяти)?
3. Какой максимальный объем памяти можно адресовать с помощью шестнадцати-, тридцатидвух- и шестидесятичетырехразрядного адресного слова?
4. Каким образом в шестнадцатиразрядной ЭВМ может быть организована возможность формирования адреса для работы с оперативным запоминающим устройством объемом 1 Мбайт (сколько разрядов в адресе для этого требуется)?
5. Каким образом осуществляется отображение страниц логического адресного пространства программы пользователя на страницы физической памяти?
6. Для чего используются регистры адреса страниц (сегментные регистры) процессора?
7. Что такое и для чего используются страничные таблицы?
8. Что такое свопинг?