

## Структура микропроцессорной системы

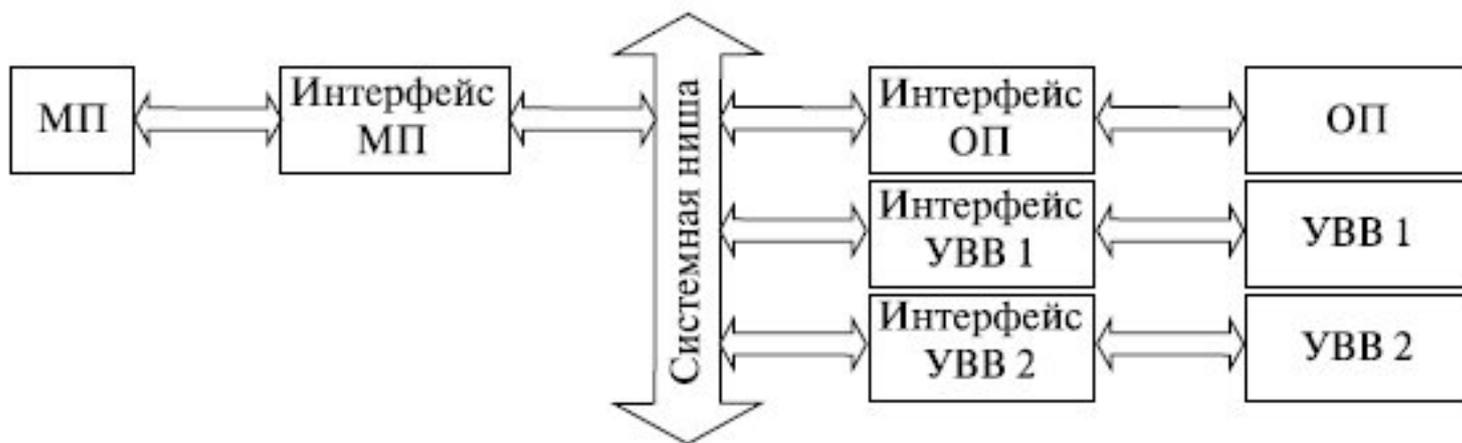
МПС - сложная система, включающая в себя большое количество различных устройств. Основой ее является *микропроцессор*. Связь устройств ЭВМ между собой осуществляется с помощью сопряжений, которые в вычислительной технике называются интерфейсами.

**Интерфейс** - это совокупность программных и аппаратных средств, предназначенных для передачи информации между компонентами ЭВМ и включающих в себя электронные схемы, линии, шины и сигналы адресов, данных и управления, алгоритмы передачи сигналов и правила интерпретации сигналов устройствами.

В широком смысле *интерфейс* включает также механическую часть (совместимость по типоразъемам) и вспомогательные схемы, обеспечивающие электрическую совместимость устройств по уровням логических сигналов, входным и выходным токам и т. д.

Подробное изучение интерфейсов и системных шин не входит в задачи данного курса. Поэтому эти вопросы мы будем рассматривать лишь с точки зрения общего представления об организации работы **микропроцессорной системы** и принципах взаимодействия составляющих ее устройств.

Основным способом организации МПС является магистрально-модульный (рис. 8.1): все устройства, включая и *микропроцессор*, представляются в виде модулей, которые соединяются между собой общей *магистралью*. Обмен информацией по магистрали удовлетворяет требованиям некоторого общего интерфейса, установленного для магистрали данного типа. Каждый *модуль* подключается к магистрали посредством специальных интерфейсных схем.



**Рис. 8.1.** Магистрально-модульный принцип построения микропроцессорной системы

На интерфейсные схемы модулей возлагаются следующие задачи:

- обеспечение функциональной и электрической совместимости сигналов и *протоколов обмена* модулей и системной магистрали;
- преобразование внутреннего формата данных модуля в формат данных системной магистрали и обратно;
- обеспечение восприятия единых команд обмена информацией и преобразование их в последовательность внутренних управляющих сигналов.

Эти интерфейсные схемы могут быть достаточно сложными. Обычно они выполняются в виде специализированных микропроцессорных БИС. Такие схемы принято называть **контроллерами**.

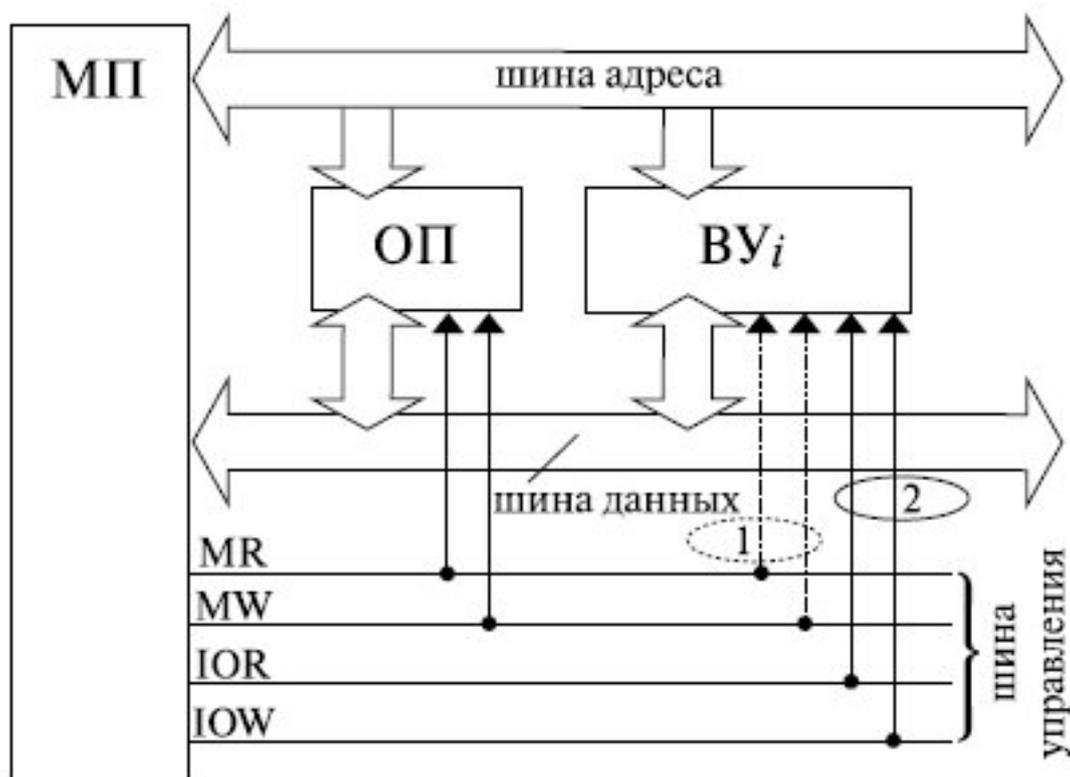
**Контроллеры** обладают высокой степенью автономности, что позволяет обеспечить параллельную во времени работу *периферийных устройств* и выполнение программы обработки данных микропроцессором.

Кроме того, предварительно буферизуя данные, контроллеры обеспечивают пересылку сразу для многих слов, расположенных по подряд идущим адресам, что позволяет использование так называемого "взрывного"

(*burst*) режима работы шины - 1 цикл адреса и следующие за ним многочисленные циклы данных.

Недостатком магистрально-модульного способа организации ЭВМ является невозможность одновременного взаимодействия более двух модулей, что ставит ограничение на *производительность* компьютера.

Взаимодействие микропроцессора с оперативной памятью (ОП) и внешними устройствами (ВУ) проиллюстрировано на рис. 8.2.



**Рис. 8.2.** Взаимодействие микропроцессора с оперативной памятью и внешними устройствами

Микропроцессор формирует адрес внешнего устройства или ячейки оперативной памяти и вырабатывает управляющие сигналы - либо **IOR/IOW** при обращении на чтение/запись из внешнего устройства, либо **MR/MW** для чтения/записи из оперативной памяти.

Для обмена информацией с внешними устройствами в МП имеются только 2 команды:

- команда ввода **IN AX, DX** записывает в регистр **AX** число из внешнего устройства, адрес которого находится в регистре **DX** ; при этом вырабатывается сигнал **IOR** (**I**Nput/**O**UTput **R**ead ).);
- команда вывода **OUT DX, AX** выводит информацию из регистра **AX** во внешнее устройство, адрес которого находится в регистре **DX** ; при этом вырабатывается сигнал **IOW** (**I**Nput/**O**UTput **W**rite ).

Сигналы **IOR/IOW** формируются при выполнении только этих команд.

Формирование сигналов **MR/MW** происходит в командах, для которых *операнд* и/или приемник результата располагаются в оперативной памяти, например, **ADD [EBX], AX**.

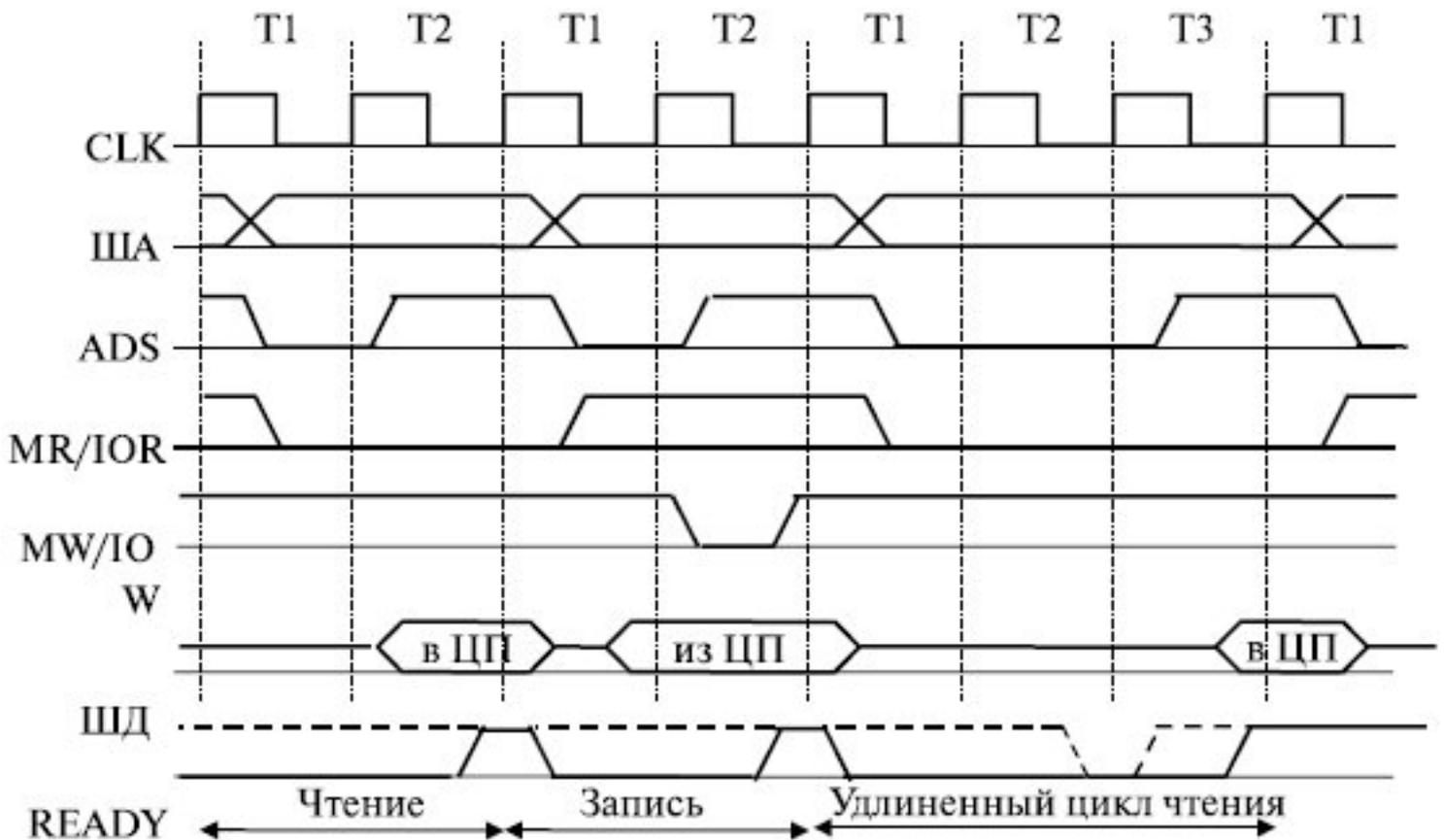
В связи с этим возможны два основных способа организации адресного пространства **микропроцессорной системы**:

1. с *общим адресным пространством* внешних устройств и оперативной памяти;
2. с *независимыми адресными пространствами*.

В первом случае к портам ввода/вывода можно обращаться как к ячейкам оперативной памяти. Достоинством такого подхода является возможность использовать различные режимы адресации при обращении к внешним устройствам, а также выполнять над содержимым портов ввода/вывода различные арифметико-логические *операции*. Но в то же время при этом сокращается емкость адресуемой ОП и снижается защищенность системы, так как она лишается дополнительных средств защиты, связанных с выполнением команд ввода/вывода (не работает *поле IORL регистра флагов*). К тому же нарушение в логике работы программы (формирование неверного адреса оперативной памяти) может привести к ложному сбратыванию внешнего устройства.

Если первый недостаток не столь существен при современных объемах *запоминающих устройств*, то второй может весьма негативно сказаться на работе МПС. Возможность использования сложных режимов адресации при обращении к внешним устройствам для микропроцессорных систем на основе универсальных МП не столь важна. Поэтому в настоящее время при построении МПС предпочтение отдается второму подходу.

Рассмотрим особенности обмена информацией микропроцессора с внешними устройствами. Упрощенная временная *диаграмма* этого процесса представлена на рис. 8.3.



**Рис. 8.3.** Временная диаграмма обмена информацией между микропроцессором и внешним устройством

При таком обмене *микропроцессор* использует стандартный цикл шины длительностью 2 периода *тактового генератора*, начало которого определяется сигналом *ADS* шины.

Однако если в обмене задействованы относительно медленное *внешнее устройство* или *оперативная память*, которые не успевают завершить обмен в текущем цикле работы микропроцессора, то они сбрасывают сигнал *READY*, который проверяется микропроцессором в конце второго *тактового*

сигнала цикла. При  $READY = 0$  МП вставляет дополнительные такты ожидания в цикл шины, чтобы дать возможность внешнему устройству завершить операцию ввода/вывода. Однако сигнал неготовности внешнего устройства не может длиться более установленного времени (для шины ISA - 2,5 мкс), в противном случае генерируется прерывание по ошибке ввода/вывода. Это предотвращает зависание системы в случае, например, механического повреждения данного контакта.

В общем случае стандартный цикл шины при обмене уже включает несколько дополнительных тактов ожидания. При этом циклы обращения к памяти короче циклов обращения к портам ввода/вывода. Минимальная длительность цикла определяется чипсетом и может программироваться опциями *BIOS Setup* заданием тактов ожидания. Если ОП или внешнее устройство не нуждаются в дополнительных тактах, то их контроллеры выставляют на шину сигнал **OWS**, по которому дополнительные такты не вставляются.

### Прямой доступ к памяти

В МПС используются два основных способа организации передачи данных между памятью и периферийными устройствами: **программно управляемая передача и прямой доступ к памяти ПДП** (*Direct Memory Access - DMA*).

**Программно управляемая передача** данных осуществляется при непосредственном участии и под управлением процессора. Например, при пересылке блока данных из внешнего устройства в оперативную память процессор должен выполнить следующую последовательность шагов:

1. сформировать начальный адрес области обмена ОП;
2. занести длину передаваемого массива данных в один из своих внутренних регистров, который будет играть роль счетчика;
3. выдать команду чтения информации из ВУ; при этом на шину адреса из МП выдается адрес ВУ, на шину управления - сигнал чтения данных из ВУ, а считанные данные заносятся во внутренний регистр МП;
4. выдать команду записи информации в ОП; при этом на шину адреса из МП выдается адрес ячейки оперативной памяти, на шину управления - сигнал записи данных в ОП, а на шину данных выставляются данные из регистра МП, в который они были помещены при чтении из ВУ;
5. модифицировать регистр, содержащий адрес оперативной памяти;
6. уменьшить счетчик длины массива на длину переданных данных;
7. если переданы не все данные, то повторить шаги 3-6, в противном случае закончить обмен.

Как мы видим, программно управляемый обмен ведет к нерациональному использованию мощности микропроцессора, который вынужден выполнять большое количество относительно простых операций, приостанавливая работу над основной программой. При этом действия, связанные с обращением к оперативной памяти и к внешнему устройству, обычно требуют удлиненного цикла работы микропроцессора, что приводит к еще более существенным потерям производительности.

Альтернативой программно управляемому обмену служит **прямой доступ к памяти** - способ быстродействующего подключения внешнего устройства, при котором оно обращается к оперативной памяти, не прерывая работы процессора. Такой обмен происходит под управлением отдельного устройства - контроллера прямого доступа к памяти (КПДП).

Схема включения КПДП в состав микропроцессорной системы представлена на рис. 8.4.

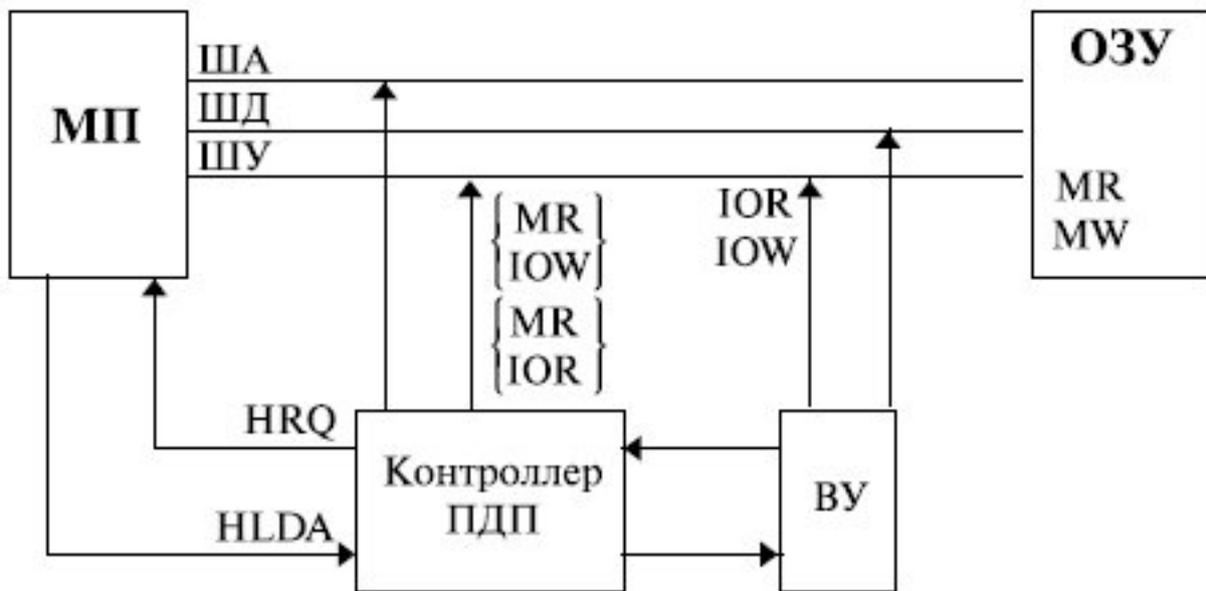


Рис. 8.4. Структура МПС с контроллером прямого доступа к памяти

Перед началом работы **контроллер ПДП** необходимо инициализировать: занести начальный *адрес* области ОП, с которой производится обмен, и длину передаваемого массива данных. В дальнейшем по сигналу запроса прямого доступа *контроллер* фактически выполняет все те действия, которые обеспечивал *микропроцессор* при программно управляемой передаче.

Последовательность действий КДПДП при запросе на *прямой доступ* к памяти со стороны внешнего устройства следующая:

1. Принять запрос на ПДП (сигнал **DRQ**) от ВУ.
2. Сформировать запрос к МП на захват шин (сигнал **HRQ**).
3. Принять сигнал от МП (**HLDA**), подтверждающий факт перевода микропроцессором своих шин в третье состояние.
4. Сформировать сигнал, сообщающий устройству ввода-вывода о начале выполнения циклов прямого доступа к памяти (**DACK**).
5. Сформировать на *шине адреса* компьютера адрес ячейки памяти, предназначенной для обмена.
6. Выработать сигналы, обеспечивающие управление обменом (**IOR**, **MW** для передачи данных из ВУ в оперативную память и **IOW**, **MR** для передачи данных из оперативной памяти в ВУ).
7. Уменьшить значение в счетчике данных на длину переданных данных.
8. Проверить условие окончания сеанса прямого доступа (обнуление счетчика данных или снятие сигнала запроса на ПДП). Если условие окончания не выполнено, то изменить адрес в регистре текущего адреса на длину переданных данных и повторить шаги 5-8.

*Прямой доступ* к памяти позволяет осуществлять обмен данными между внешним устройством и оперативной памятью параллельно с выполнением процессором программы.

Структура КДПДП представлена на рис. 8.5.

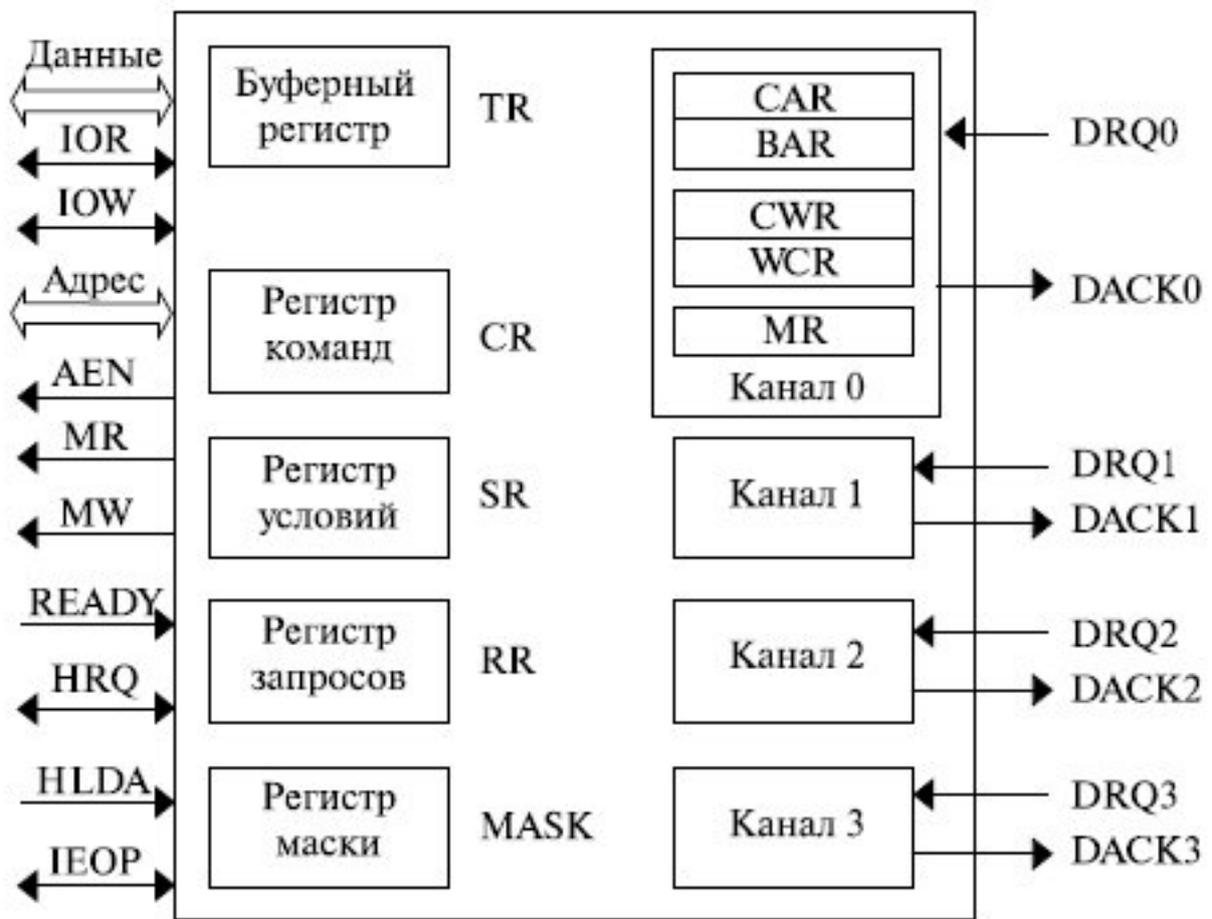


Рис. 8.5. Структура контроллера прямого доступа к памяти

Контроллер состоит из 4 идентичных каналов, позволяющих подключить к системе до четырех устройств, работающих в режиме ПДП.

В состав каждого канала входят следующие регистры:

- **MR ( Mode Register )** - регистр режима. Определяет следующие параметры передачи:
  - порядок изменения (увеличения или уменьшения) адреса ОП при передаче;
  - возможность автоинициализации;
  - режим обслуживания:
    - одиночная передача (контроллер возвращает магистраль процессору после каждого цикла ПДП);
    - блочная передача (контроллер владеет магистралью в течение передачи всего массива);
    - по требованию (окончание передачи определяется снятием сигнала DRQ или подачей сигнала IEOP на внешний вход КПДП);
    - каскадирование;
- **CAR ( Current Address Register )** - регистр текущего адреса;
- **BAR ( Base Address Register )** - базовый регистр адреса;
- **CWR ( Current Word Register )** - текущий счетчик данных;
- **WCR ( W0ord ConTRol Register )** - базовый счетчик данных.

Значения в регистрах **BAR** и **WCR** устанавливаются при инициализации и в ходе циклов ПДП не меняются. В регистры **CAR** и **CWR** в начале выполнения ПДП заносятся значения из регистров **BAR** и **WCR** соответственно. При выполнении ПДП эти регистры изменяются.

Управляющие регистры, общие для всего контроллера:

- **CR ( Command Register )** - регистр команд - определяет:

- режим память-память или обычный. В режиме память-память осуществляется обмен по схеме ПДП между двумя областями ОП (только для каналов 0 и 1) с использованием буферного регистра TR ( *Temporary Register* );
- запрет/разрешение ПДП;
- порядок изменения приоритетов каналов:
- фиксированный приоритет;
- циклическое изменение приоритета после обработки запроса на ПДП по одному из каналов;
- явное указание наиболее приоритетного канала;
- уровень сигналов DRQ и DACK (настройка на активный уровень сигналов под особенности работы внешних устройств);
- SR ( *Status Register* ) - регистр условий: используется для фиксации признаков обмена по каждому из каналов и программных запросов на ПДП.

Во избежание ложных срабатываний внешних устройств, не использующих в данный момент режим прямого доступа, контроллер ПДП во время режима ПДП вырабатывает сигнал AEN, который блокирует работу остальных внешних устройств:

$$AEN = \begin{cases} 1\text{-шиной управляет контроллер ПДП} \\ 0\text{-шиной управляет МП или другой задатчик} \end{cases}$$

Для увеличения количества внешних устройств, которые могут быть подключены к микропроцессору в режиме ПДП, используется каскадное включение КПДП (рис. 8.6).

При этом ведомый КПДП подключается к одному из каналов ведущего контроллера по схеме подключения внешнего устройства, а непосредственно с микропроцессором связывается только ведущий контроллер. В стандартной конфигурации персональной ЭВМ применяются два КПДП (ведущий и ведомый), которые позволяют подключить до 7 внешних устройств, причем 2 канала закреплены за накопителями на жестком и гибком дисках.

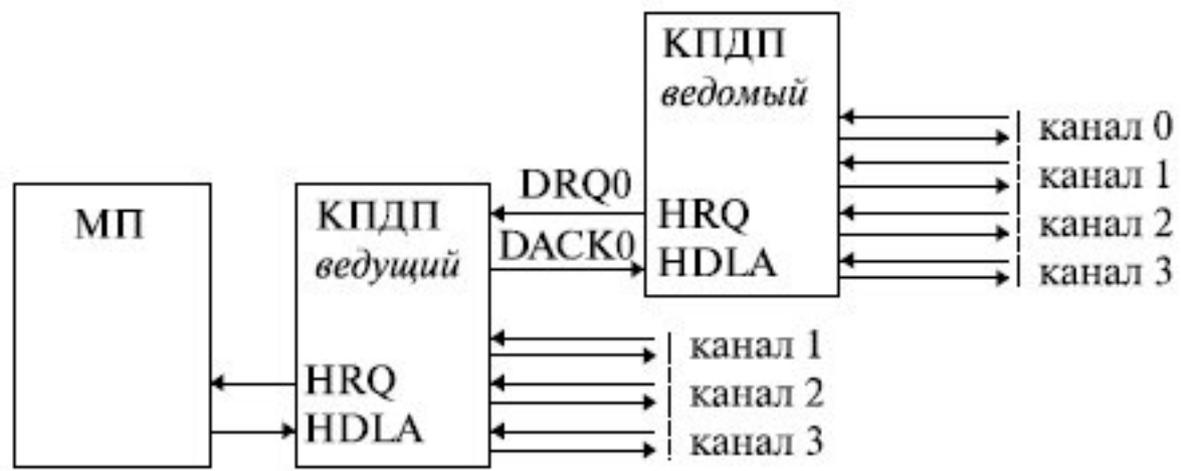


Рис. 8.6. Каскадное включение контроллеров прямого доступа к памяти

В максимальной конфигурации при использовании ведущего и 4 ведомых КПДП, подключенных к каждому из каналов ведущего, к МП можно подключить до 16 внешних устройств, работающих в режиме ПДП.

### Назначение и функции чипсета в микропроцессорной системе

**Чипсет (chIPset)** - это набор БИС (обычно 1-3 микросхемы), функционально эквивалентный микросхемам, входящим в стандартную конфигурацию микропроцессорной системы.

Как правило, чипсет интегрирует в себе функции следующих устройств:

- контроллера оперативной памяти;
- контроллеров кэш-памяти 2-го и/или 3-го уровня;
- контроллеров ПДП;
- контроллеров приоритетных прерываний;
- контроллера клавиатуры;
- контроллера мыши PS/2 ;
- контроллера инфракрасного порта;
- таймера реального времени;
- моста шины PCI ;
- моста шины ISA и др.

Обычно в составе чипсета выделяют:

- северный мост (North Bridge) - системный контроллер, в который входит контроллер системной шины, шин AGP и PCI, ОЗУ и кэш-памяти (для наборов под обычный Pentium);
- южный мост (SOUTH Bridge) - периферийный контроллер, включающий контроллеры EIDE, клавиатуры, моста PCI-to-PCI, последовательных/параллельных портов, шины USB и других подобных устройств.

Выбор чипсета во многом определяет конфигурацию МПС и ее производительность. Если МП можно заменить, а емкость ОЗУ увеличить, то замена чипсета однозначно связана с заменой системной платы, а ограничения чипсета также однозначно ограничивают возможности замены других элементов МПС: МП, ОЗУ, внешних устройств.

Чипсет накладывает ограничения на следующие функциональные характеристики системы в целом: тип памяти, тип кэш-памяти второго и/или третьего уровня, тип МП, максимальная частота системной шины, тип шины PCI (32- или 64-разрядная); поддержка многопроцессорной конфигурации и некоторые другие характеристики.

Практика показывает, что разница в производительности системных плат разных фирм, построенных с применением одного и того же чипсета, составляет от силы несколько процентов, между тем как тот же параметр для различных чипсетов может отличаться на порядок.

Рассмотрим использование чипсета на примере организации микропроцессорной системы на базе МП Pentium III на ядре Katmai (рис. 8.7).

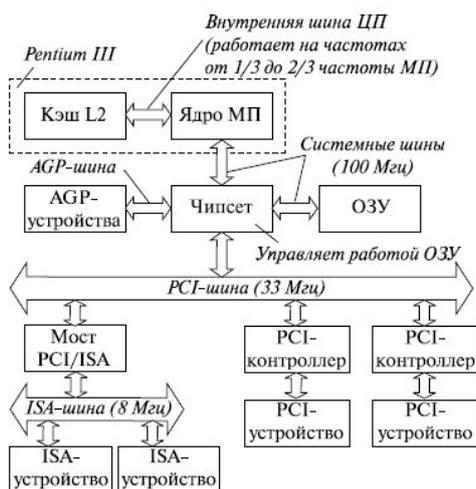


Рис. 8.7. Структура микропроцессорной системы с использованием чипсета

Частота шины *PCI* составляет 33 МГц и при частоте системной шины и 66, и 100 МГц (используется деление частоты на 2 и 3 соответственно). *AGP (Accelerated Graphics Port)* - 32-разрядная шина (тактовая частота всегда 66 МГц), используемая для подключения графического адаптера и позволяющая ускорить обмен данными между графическим адаптером и основной памятью. В режиме *AGP 2x* эффективная частота в 133

МГц обеспечивается использованием обоих фронтов импульсов частоты 66 МГц. Шина *ISA* работает на частоте 8 МГц. За ее функционирование отвечает мост *PCI/ISA*, являющийся обычным *PCI*-устройством (просто встроенным в чипсет).

Системная шина может работать на разной частоте, но все эти варианты совместимы по разводке и электрически, что создает почву для "разгона".

Таким образом, при смене частоты системной шины, например, с 66 на 100 МГц замене подлежат лишь три устройства: процессор, чипсет и память. Никакие другие устройства не заметят подмены шины - ни *AGP* видеоадаптеры, ни любой из *PCI*-контроллеров, ни старые *ISA*-карты. В этом и заключается принципиальное отличие шины 100 МГц от шины 66

МГц, разогнанной до 75/83 МГц. В первом случае новый чипсет обеспечивает работу *PCI* на частоте 33 (100/3) МГц, во втором старый чипсет и не подозревает о "разгоне" и продолжает делить частоту системной шины на 2, а в итоге частота *PCI*-шины равна 37,5 или 41,5 МГц, что может привести к неправильной работе некоторых *PCI*-устройств.