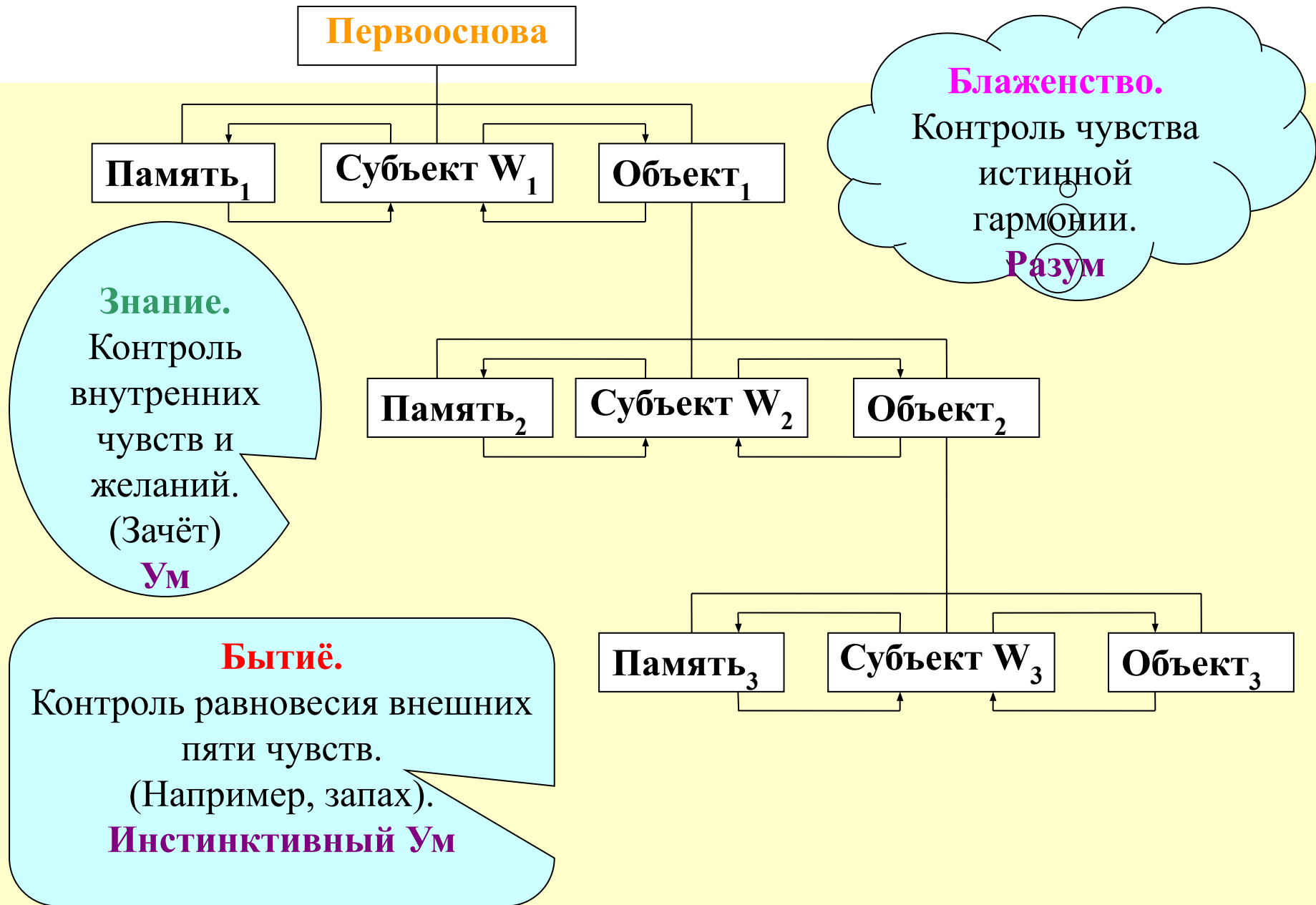


Три слоя Личности.



УМ

- Человек должен понять ум, как он работает, его возможности. Вы должны научиться управлять им, а не уступать ему.
- Ум имеет два аспекта: деятельный и воспринимающий. Он есть причина привязанности и чувства обладания.
- Ум создает представление об удовольствии и боли, радости и печали, а также все те мотивы, которые побуждают человека действовать.

Разум

- Выше инстинктивного ума стоит интеллект. Он анализирует и классифицирует впечатления, собранные умом с помощью органов чувств, принимает решения. Чувства - это инструменты, используемые для контакта с Природой и сбора информации об Ум имеет склонность к разделению и видит мир, в многообразии. Разум пытается улавливать единство в многообразии. Разум подчинен сознанию . А ум должен быть подчинен разуму.
- Но на самом деле он служит чувствам, которые должны быть только его слугами. Таким образом, ум порабощает человека

- Разум находится очень близко к *Первоисточнику* и поэтому может обрести до 90% Его энергии и света. Ум черпает свою силу в Разуме, чувства - в уме, а тело в чувствах. В этом ступенчатом процессе истечения силы от *Первоисточника (Атмы)* к телу происходит количественное снижение и качественное изменение (загрязнение) энергии. Чистота и сила скрытого *Первоисточника* постепенно снижаются по мере того, как она проходит через разум, ум, органы чувств и достигает тела.

Лек№12. Подсистема ввода/вывода

Технически система ввода/вывода в рамках ВМ реализуется комплексом *модулей ввода/вывода* (МВВ). Модуль ввода/вывода выполняет сопряжение ВУ с ядром ВМ и различные коммуникационные операции между ними. Две основные функции МВВ

- обеспечение интерфейса с ЦП и памятью (*«большой» интерфейс*);
- обеспечение интерфейса с одним или несколькими периферийными устройствами (*«малый» интерфейс*).

Анализируя архитектуру известных ВМ, можно выделить три основных способа подключения СВВ к ядру процессора (рис. 8.1).

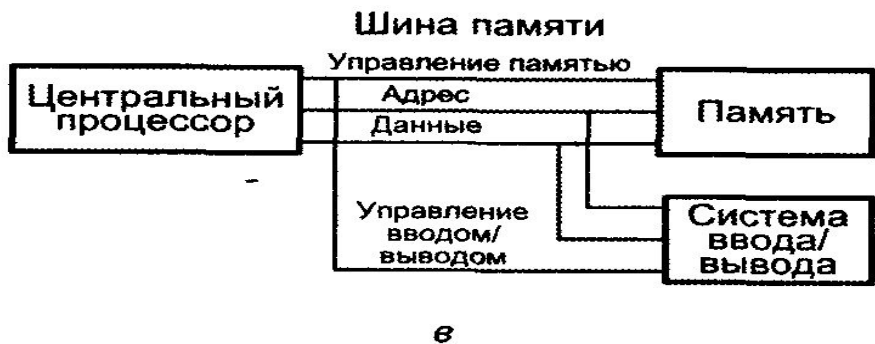
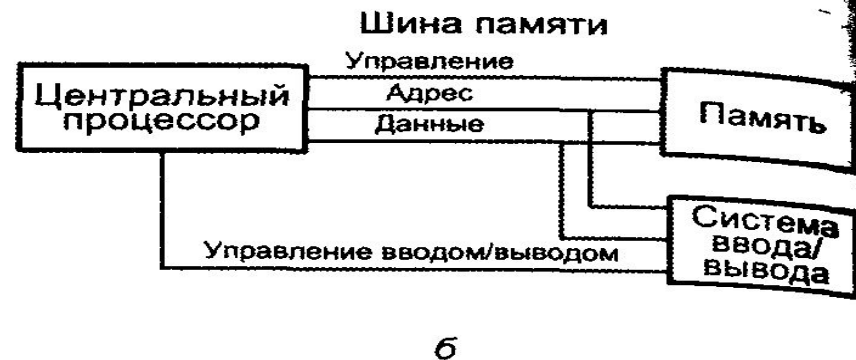


Рис. 8.1. Место системы ввода/вывода в архитектуре вычислительной машины: а — раздельными шинами памяти и ввода/вывода; б — с совместно используемыми линиями данных и адреса, в — подключение на общих правах с процессором и памятью

а) С раздельными шинами памяти и ввода/вывода. Возможность осуществлять обращение к памяти одновременно с выполнением ввода/вывода, шина в/в может иметь меньшую пропускную способность, что позволяет снизить затраты. Недостаток - большое количество точек подключения к ЦП

б) Память и СВВ имеют общие линии адреса и данных, разделяемых во времени. Управление памятью и СВВ и синхронизация их взаимодействия с процессором осуществляются по раздельным линиям. Учитываются особенности процедур обращения к памяти и к модулям ввода/вывода для эффективности доступа.

в) Предполагает подключение СВВ к системной шине на общих правах с процессором и памятью. Возрастают унификации соединения, конфликтов на шине, сложности управления.

а) С отдельными шинами памяти и ввода/вывода. Возможность осуществлять обращение к памяти одновременно с выполнением ввода/вывода, шина в/в может иметь меньшую пропускную способность, что позволяет снизить затраты. Недостаток - большое количество точек подключения к ЦП

б). Память и СВВ имеют общие линии адреса и данных, разделяемых во времени. Управление памятью и СВВ и синхронизация их взаимодействия с процессором осуществляются по отдельным линиям. Учитываются особенности процедур обращения к памяти и к модулям ввода/вывода для эффективности доступа.

в) Предполагает подключение СВВ к системной шине на общих правах с процессором и памятью. Возрастают унификации соединения, конфликтов на шине, сложности управления.

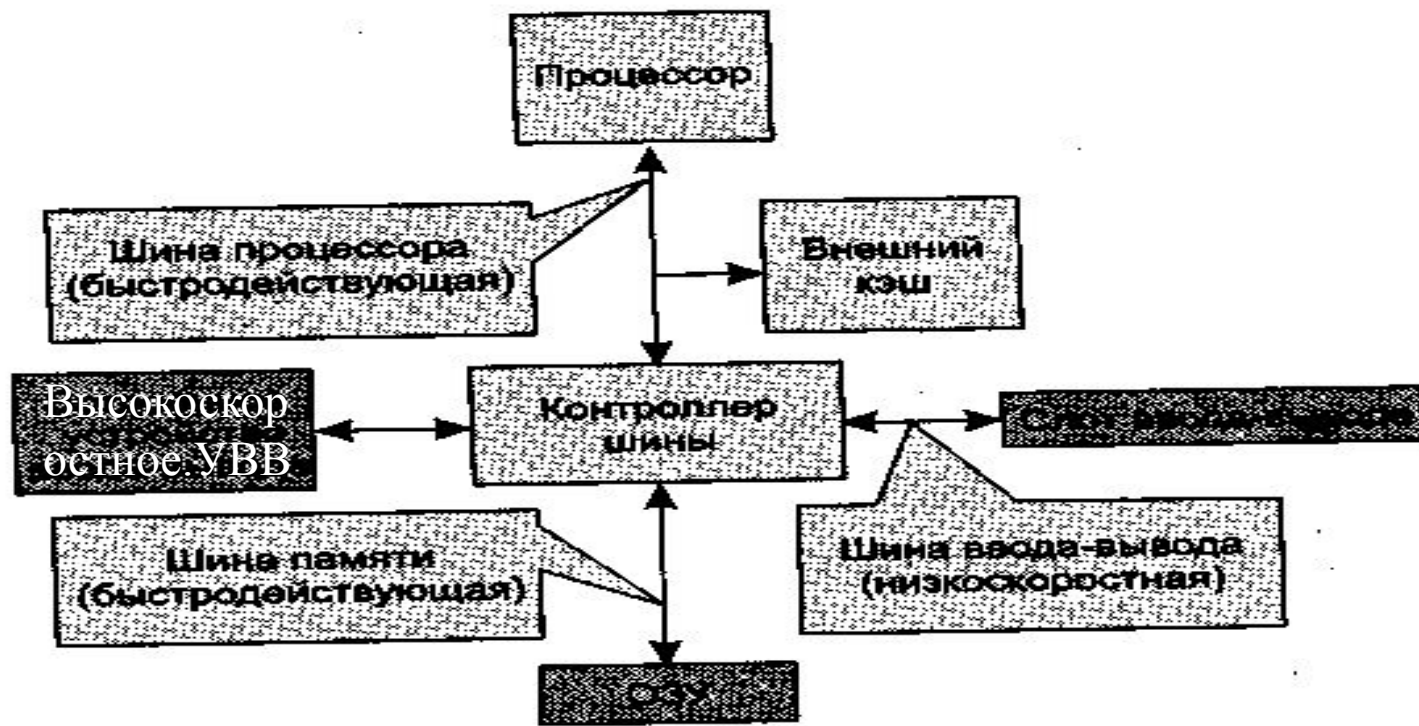


Рис. 10.1. Конфигурация системы с шиной ISA

- Шина ISA (Industry Standard Architecture — промышленная стандартная архитектура) представлена в двух версиях: для IBM PC/XT (1981 год) и для PC AT (1984 год). Она использовалась в первом компьютере IBM PC, и тогда это было неофициальное название шины IBM PC/XT, позволяющей добавлять в систему различные устройства.
- ISA — основная шина на устаревших материнских платах. С появлением 32-разрядных высокоскоростных МП шина ISA стала существенным препятствием увеличения быстродействия ПК.

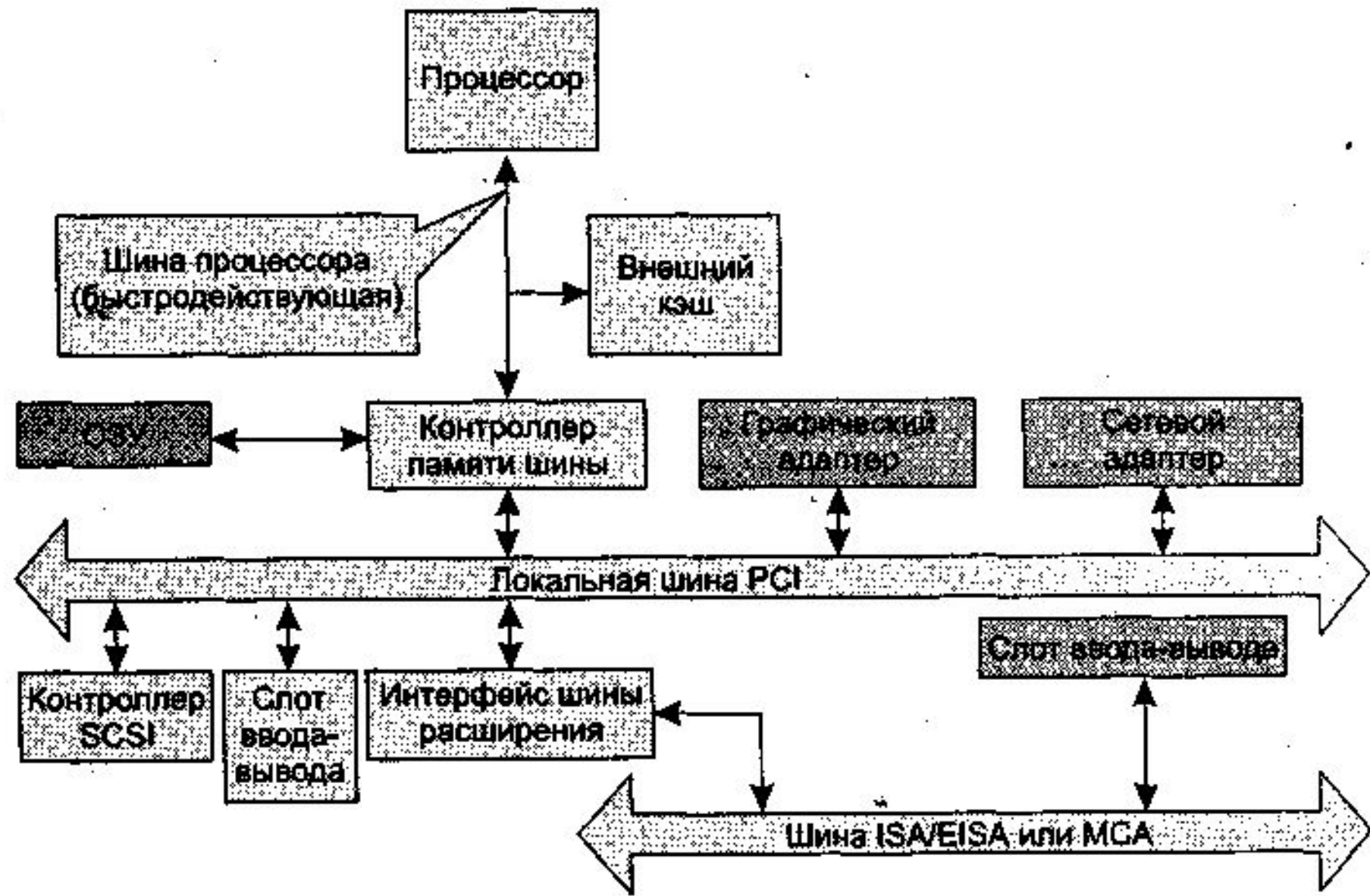


Рис. 10.2. Конфигурация системы с шиной PCI

- Шина *PCI* (Peripheral Component Interconnect, соединение периферийных компонентов) — самый распространенный и универсальный интерфейс для подключения различных устройств. Базовая версия PCI 1.0 (IEEE 1.386 (Institute of Electrical and Electronic Engineers 1.386 — стандарт Института инженеров по электротехнике и электронике 1.386) разработана в 1991 году фирмой Intel с целью создать шину, способную заменить все существующие, часто несовместимые шинные интерфейсы, такие как ISA, EISA, MCA, VLB. Шина PCI 1.0 является намного более универсальной, чем VLB; допускает подключение до 10 устройств; имеет свой адаптер, позволяющий ей настраиваться на работу с любым МП от 80486 до современных Pentium. Тактовая частота PCI 1.0 — 33 МГц, разрядность — 32 разряда для данных и 32 разряда для адреса с возможностью расширения до 64 разрядов, теоретическая пропускная способность 132 Мбайт/с, а в 64-разрядном варианте — 264 Мбайт/с.

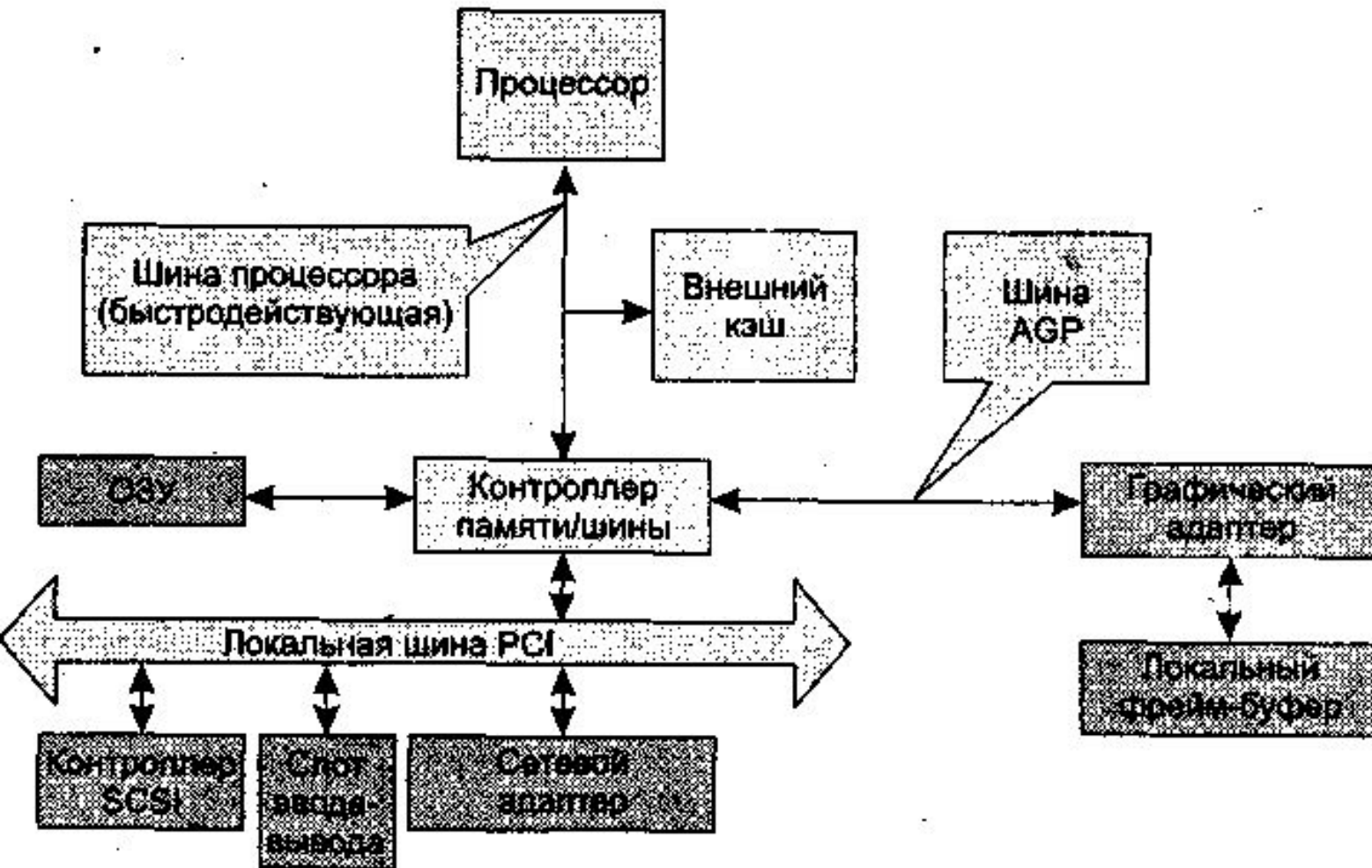


Рис. 10.3. Конфигурация системы с шиной AGP

- Шина *AGP* (Accelerated Graphics Port — ускоренный графический порт) — интерфейс для подключения видеоадаптера к отдельной магистрали AGP, имеющей выход непосредственно на системную память. Разработана шина на основе стандарта PCI v2.1.

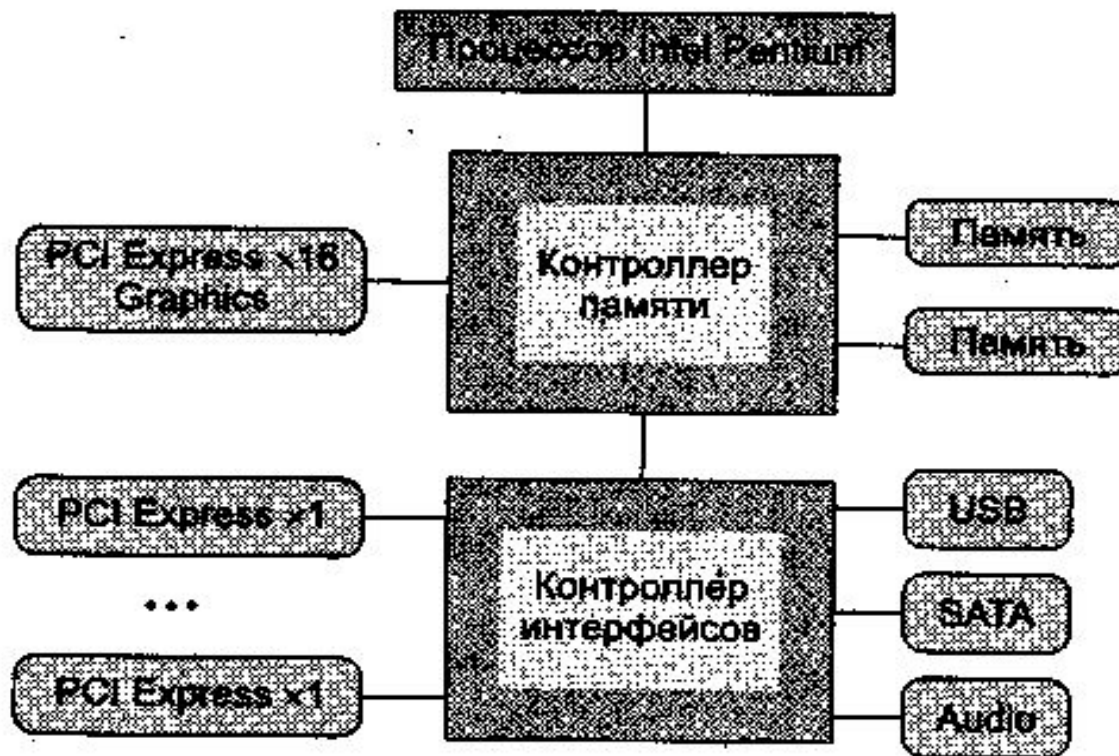


Рис. 10.4. Простейшая системная топология с интерфейсом PCI Express

- Пожалуй, наиболее перспективно и представляет существенный интерес семейство последовательных интерфейсов PCI Express, информация о базовом протоколе которого появилась в июле 2002 года. PCI Express использует совокупность независимых последовательных каналов передачи данных. Поскольку при передаче используется помехозащищенное кодирование, каждый байт представляется 10 битами. Пропускная способность одного канала 200 Мбайт/с. Лицензированы 1-, 2-, 4-, 8-, 16- и 32-канальные версии (до 6,4 Гбайт/с). В режиме дуплексной передачи все эти цифры пропускной способности

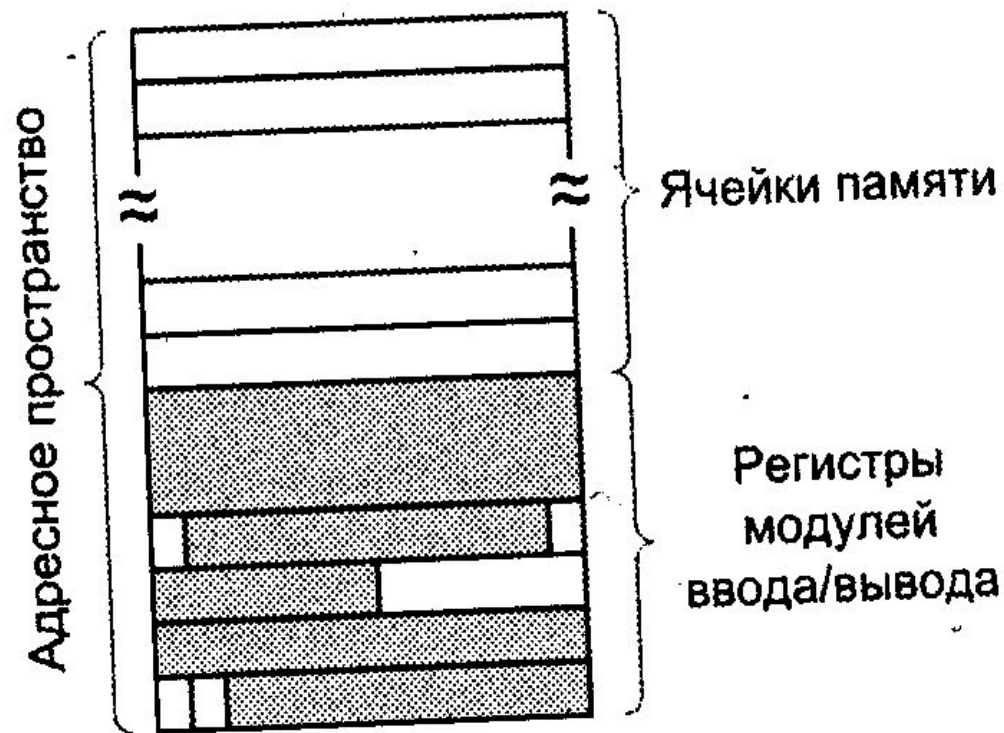


Рис. 8.2. Распределение совмещенного адресного пространства

Операции в/в предполагают наличие системы адресации, позволяющей выбрать один из модулей СВВ, а также одно из подключенных к нему внешних устройств. Адрес модуля и ВУ является частью команды, а место на внешнем устройстве определяется пересылаемой на ВУ информацией.

Адресное пространство ввода/вывода может быть совмещено с адресным пространством памяти или быть выделенным.

Достоинства совмещенного адресного пространства:

- расширение набора команд для обращения к внешним устройствам, что позволяет сократить длину программы и повысить быстродействие;
- значительное увеличение количества подключаемых внешних устройств;
- возможность внепроцессорного обмена данными между внешними устройствами, если в системе команд есть команды пересылки между ячейками памяти;
- возможность обмена информацией не только с аккумулятором, но и с любым регистром центрального процессора.

Недостатки совмещенного адресного пространства: |

- сокращение области адресного пространства памяти;
- усложнение декодирующих схем адресов в СВВ;
- трудности распознавания операций передачи информации при вводе/выводе среди других операций. Сложности в чтении и отладке программы, в которой простые команды вызывают выполнение сложных операций ввода/вывода;
- трудности при построении СВВ на простых модулях в/в: сигналы управления не смогут координировать сложную процедуру в/в. Поэтому МВВ часто должны генерировать дополнительные сигналы под управлением программы.

Совмещенное адресное пространство используется в вычислительных машинах MIPS и SPARC.

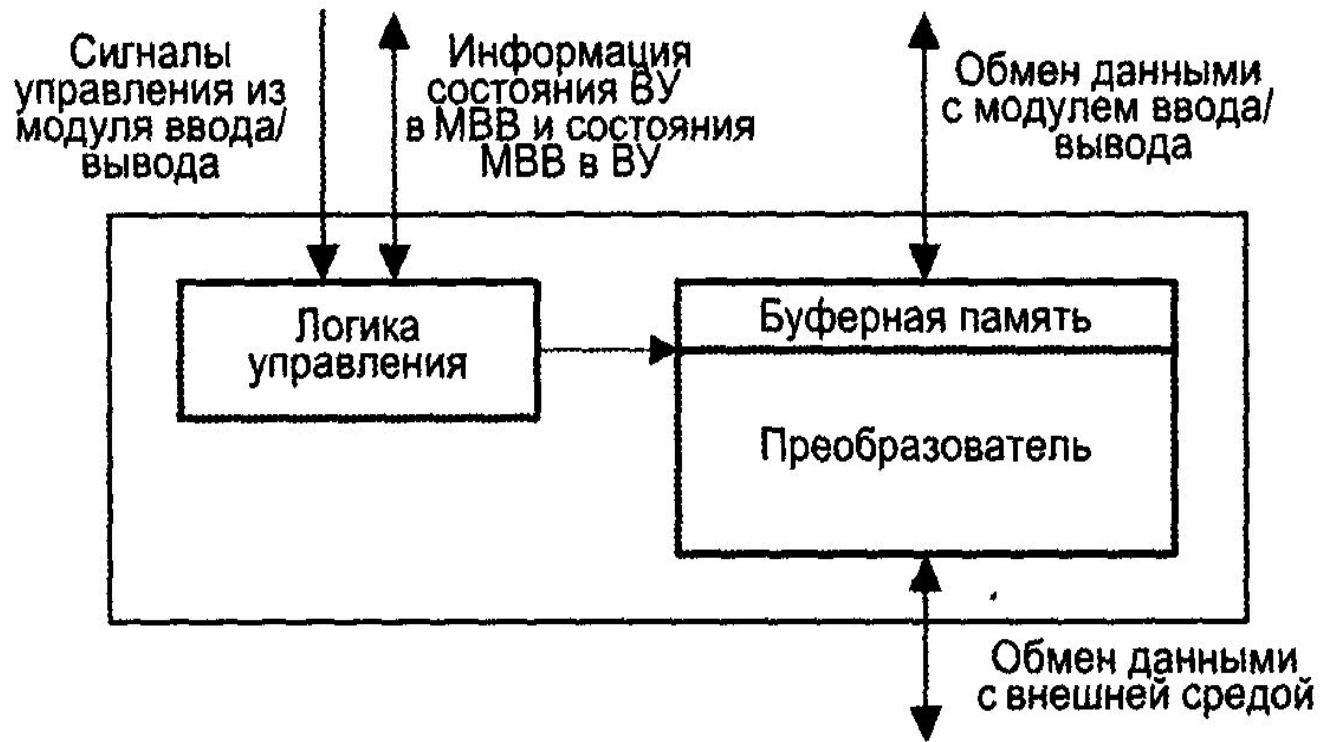


Рис. 8.3. Структура внешнего устройства

Все множество ВУ можно свести к трем категориям :

- для общения с пользователем,
- для общения с ВМ,
- для связи с удаленными устройствами.

Функции модуля ввода/вывода

Модуль в/в в составе вычислительной машины отвечает за управление одним или несколькими ВУ и за обмен данными между этими устройствами с одной стороны, и основной памятью или регистрами ЦП — с другой. Основные функции МВВ можно сформулировать следующим образом:

- локализация данных;
- управление и синхронизация;
- обмен информацией;
- буферизация данных;
- обнаружение ошибок.

Локализация данных - возможность обращения к одному из ВУ, а также адресации данных на нем. Адрес ВУ обычно содержится в адресной части команд в/в. Каждому модулю назначается определенный диапазон адресов. Старшие разряды в адресах диапазона, выделенного модулю, обычно одинаковы и обеспечивают выбор одного из МВВ в рамках системы в/в. Младшие разряды адреса представляют собой уникальные адреса регистров данного модуля или подключенных к нему ВУ.

Управление и синхронизация

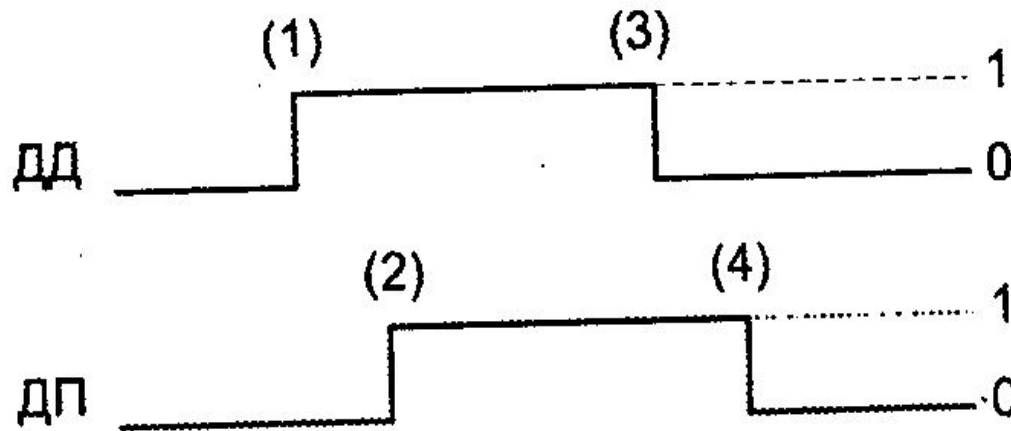


Рис. 8.4. Временная диаграмма процедуры «рукопожатия»

1. Центральный процессор с помощью сигнала $ДД = 1$ (данные достоверны) извещает о доступности данных, подлежащих выводу.
2. Приняв данные, устройство вывода сообщает процессору об их получении сигналом $ДП = 1$ (данные приняты).
3. Получив подтверждение, ЦП обнуляет сигнал $ДД$ и снимает данные с шины, после чего может выставить на шину новые данные.
4. Обнаружив, что $ДД = 0$, устройство вывода, в свою очередь, устанавливает в нулевое состояние сигнал $ДП$, после чего оно готово для обработки принятых данных все время до получения очередного сигнала $ДД = 1$.

Обмен информацией

МВВ обеспечивает обмен информацией: со стороны «большого» интерфейса - это обмен с ЦП, а со стороны «малого» интерфейса - обмен с ВУ.

Типовая последовательность операций, выполняемых процессором при вводе/ выводе:

1. Выбор требуемого внешнего устройства. -
2. Определение состояния МВВ и ВУ.
3. Выдача указания модулю ввода/вывода на подключение нужного ВУ к процессору.
4. Получение от МВВ подтверждения о подключении затребованного ВУ к процессору.
5. Распознавание сигнала готовности устройства к передаче очередной порции информации.
6. Прием (передача) порции информации.
7. Циклическое повторение двух предшествующих пунктов до завершения передачи информации в полном объеме.
8. Логическое отсоединение ВУ от процессора.

Буферизация (согласование скоростей)

Устройство	Режим работы	Партнер	<i>Скорость передачи данных, Кбайт/с</i>
Сканер	Ввод	Человек	200
Строчный принтер	Вывод	Человек	1
Лазерный принтер	Вывод	Человек	100
Граф. дисплей	Вывод	Человек	30000
Локальная сеть	Ввод/вывод	ВМ	200
Гибкий диск	Память	ВМ	50
Оптический диск	Память	ВМ	500
Магнитный диск	Память	ВМ	2000

Обнаружение ошибок

Еще одной из важнейших функций МВВ является обнаружение ошибок, возникающих в процессе ввода/вывода. Центральный процессор *следует оповещать* о каждом случае обнаружения ошибки.

Причинами возникновения последних бывают самые разнообразные факторы, которые в первом приближении можно свести к следующим группам

- воздействие внешней среды,
- старение элементной базы,
- системное программное обеспечение,
- пользовательское программное обеспечение

Из наиболее «активных» факторов окружения ВМ следует выделить

- загрязнение и влагу,
- повышенную или пониженную температуру окружающей среды,
- электромагнитное облучение,
- скачки напряжения питания.

«Большой» интерфейс

«Малый» интерфейс

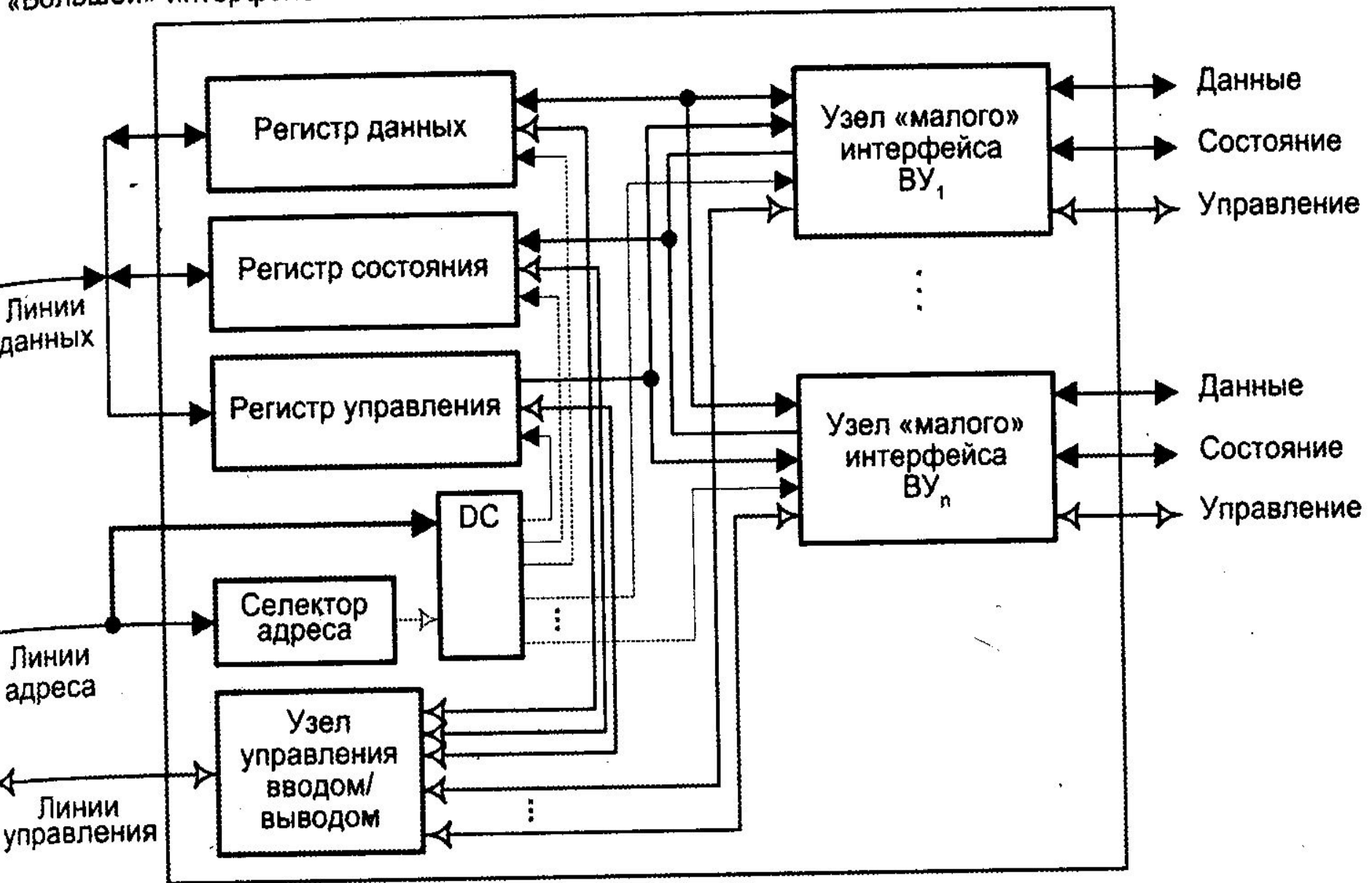


Рис. 8.5. Структура модуля ввода/вывода

Структура МВВ

- Данные, передаваемые в модуль и из него, буферизируются в регистре данных. Разрядность регистра, как правило, совпадает с шириной шины данных со стороны «большого» интерфейса (2, 4 или 8 байт) В свою очередь, **большинство В У** ориентировано на побайтовый обмен информацией. Со стороны «малого» интерфейса регистр данных дополняют узлом упаковки/распаковки (на схеме не показан). Этот узел при вводе обеспечивает последовательное побайтовое заполнение регистра данных (упаковку);
- В **регистре управления** (РУ) фиксируются поступившие из ЦП команды управления модулем или подключенными к нему внешними устройствами;
- **Регистр состояния** (РС) служит для хранения битов состояния МВВ и подключенных к нему ВУ. Содержимое определенного разряда регистра может характеризовать, например, готовность устройства ввода к приему очередной порции данных, занятость устройства вывода или нахождение ВУ в автономном режиме (offline).

Методы управления вводом/выводом

Три способа организации ввода/вывода : *программно управляемый ввод/вывод, ввод/вывод по прерываниям, прямой доступ к памяти.*

- При *программно управляемом вводе/выводе* ЦП выполняет программу, которая обеспечивает прямое управление процессом ввода/вывода, включая проверку состояния устройства, выдачу команд ввода или вывода. Выдав в МВВ команду, центральный процессор должен ожидать завершения ее выполнения, и, поскольку ЦП работает быстрее, чем МВВ, это приводит к потере времени
- *Ввод/вывод по прерываниям* во многом совпадает с программно управляемым методом. Отличие состоит в том, что после выдачи команды ввода/вывода ЦП не должен циклически опрашивать МВВ для выяснения состояния устройства. Вместо этого процессор может продолжать выполнение других команд до тех пор, пока не получит запрос прерывания от МВВ, извещающий о завершении выполнения ранее выданной команды В/ВЫВ. Как и при программно управляемом В/ВЫВ, ЦП отвечает за извлечение данных из памяти (при выводе) и запись данных в память (при вводе)
- Повышение как скорости В/ВЫВ, так и эффективности использования ЦП обеспечивает третий способ В/ВЫВ — *прямой доступ к памяти (ПДП)*. В этом режиме основная память и модуль ввода/вывода обмениваются информацией напрямую

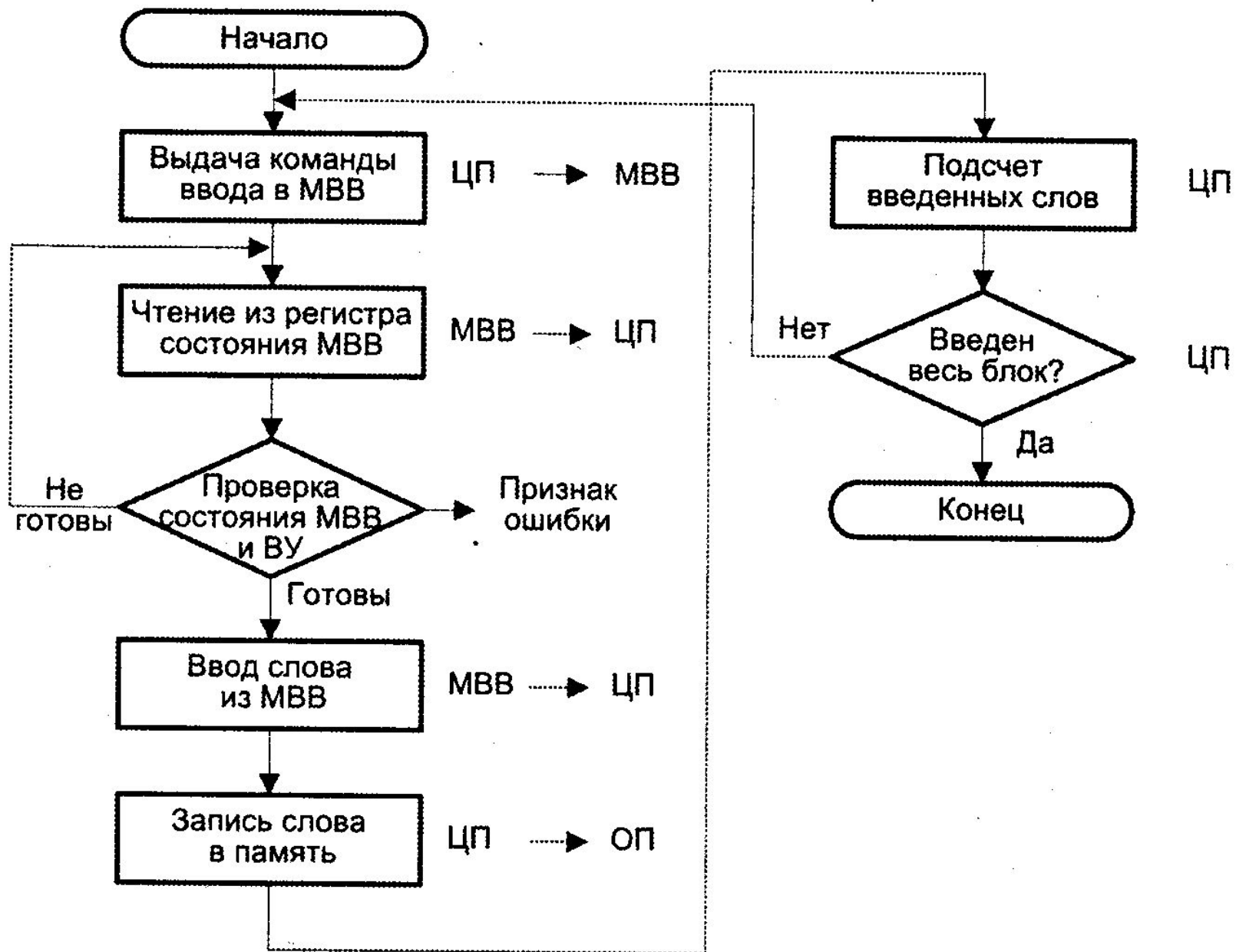


Рис. 8.6. Программно управляемый ввод данных

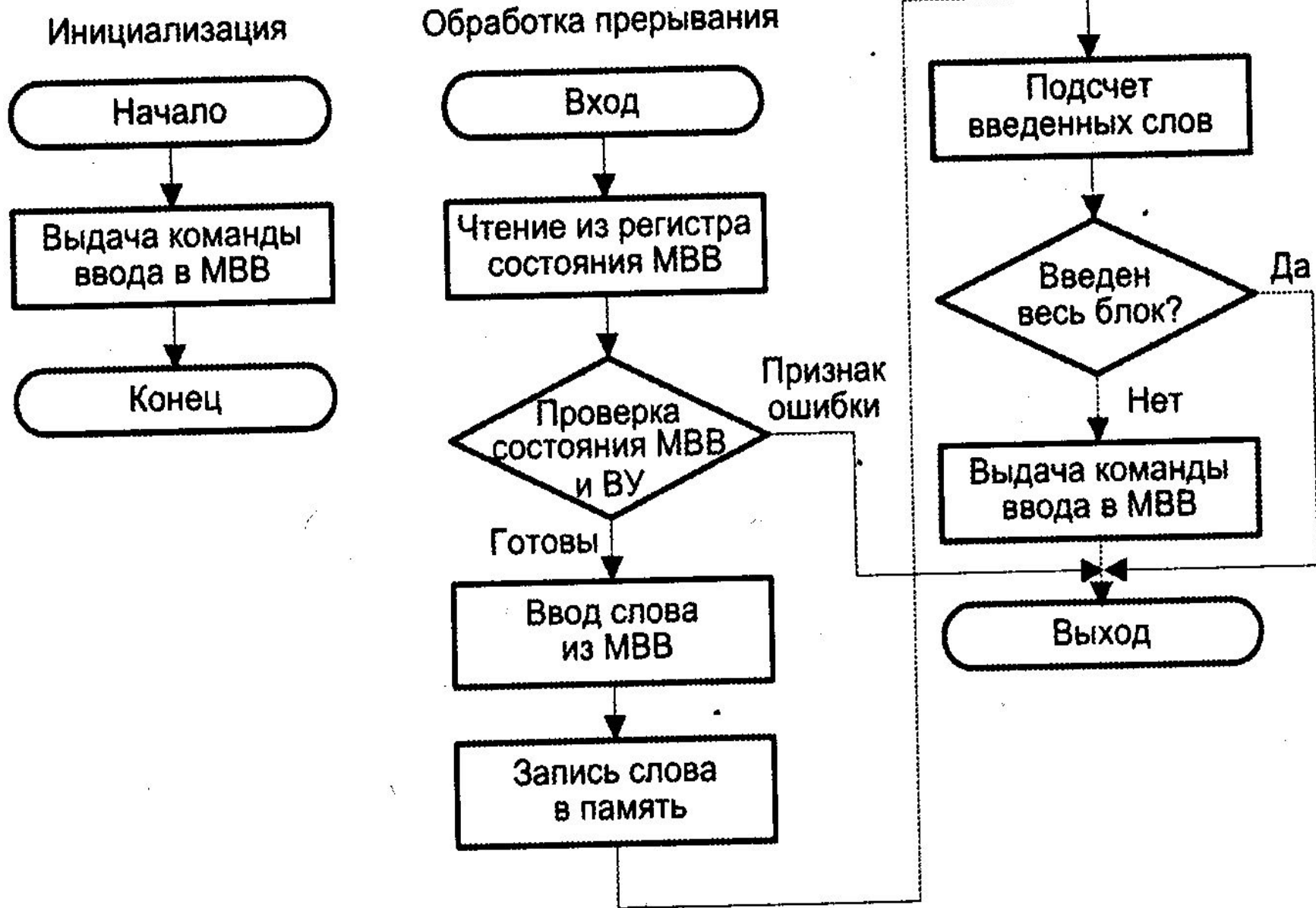


Рис. 8.7. Ввод данных по прерыванию

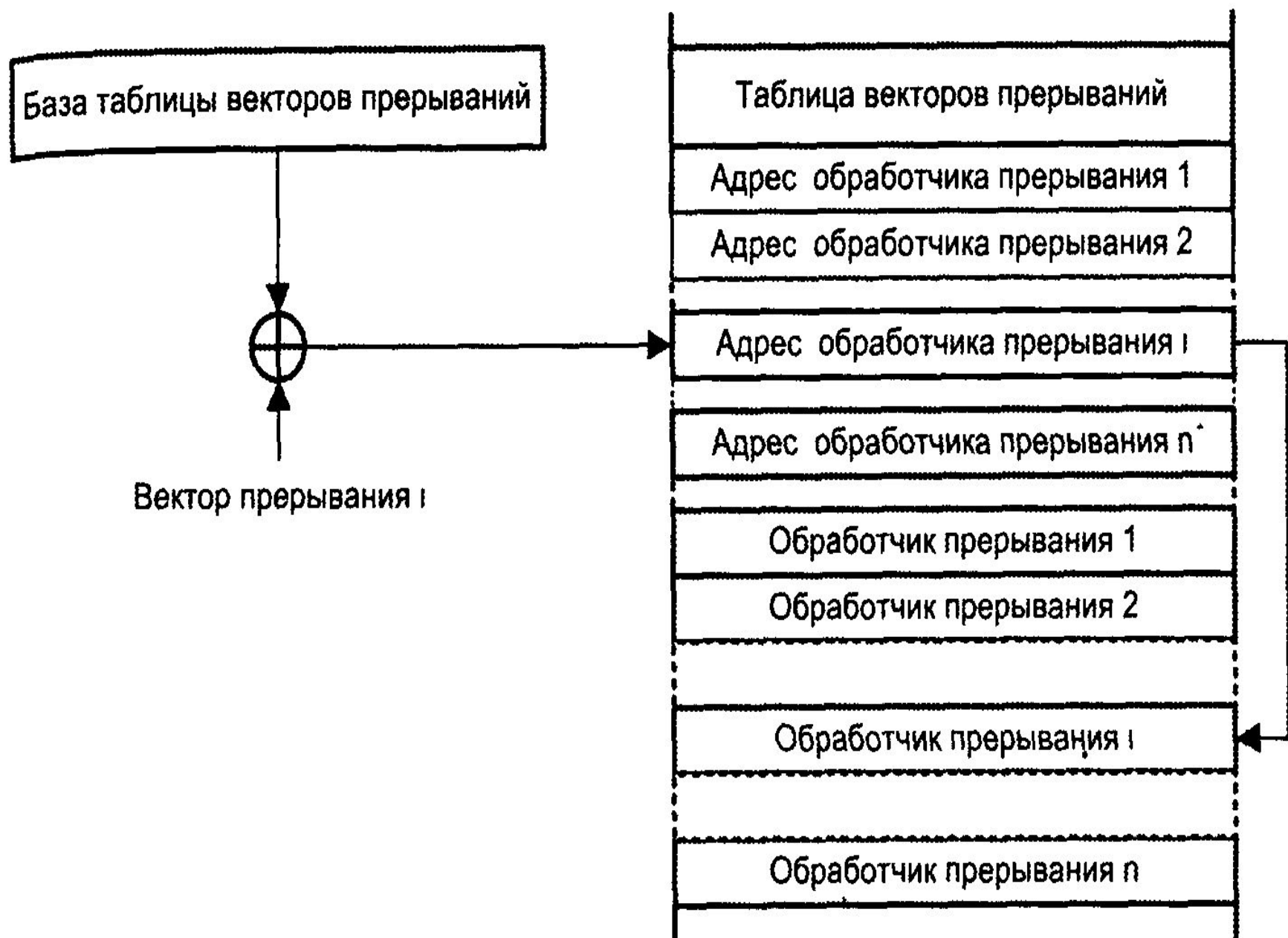


Рис. 8.8. Идентификация запроса с помощью вектора прерывания

Прямой доступ к памяти

- Когда пересылаются большие объемы данных, требуется более эффективный способ ввода/вывода — *прямой доступ к памяти* (ПДП). ПДП предполагает наличие на системной шине дополнительного модуля — *контроллера прямого доступа к памяти* (КПДП), способного брать на себя функции ЦП по управлению системной шиной и обеспечивать прямую пересылку информации между ОП и ВУ, без участия центрального процессора. В сущности, КПДП — это и есть модуль ввода/вывода, реализующий режим прямого доступа к памяти.
- Если ЦП желает прочитать или записать блок данных, он прежде всего должен поместить в КПДП (рис. 8.9) информацию, характеризующую предстоящее действие. Этот процесс называется инициализацией КПДП и включает в себя занесение в контроллер следующих четырех параметров:
 - вида запроса (чтение или запись);
 - адреса устройства ввода/вывода;
 - адреса начальной ячейки блока памяти, откуда будет извлекаться или куда будет вводиться информация;
 - количества слов, подлежащих чтению или записи.

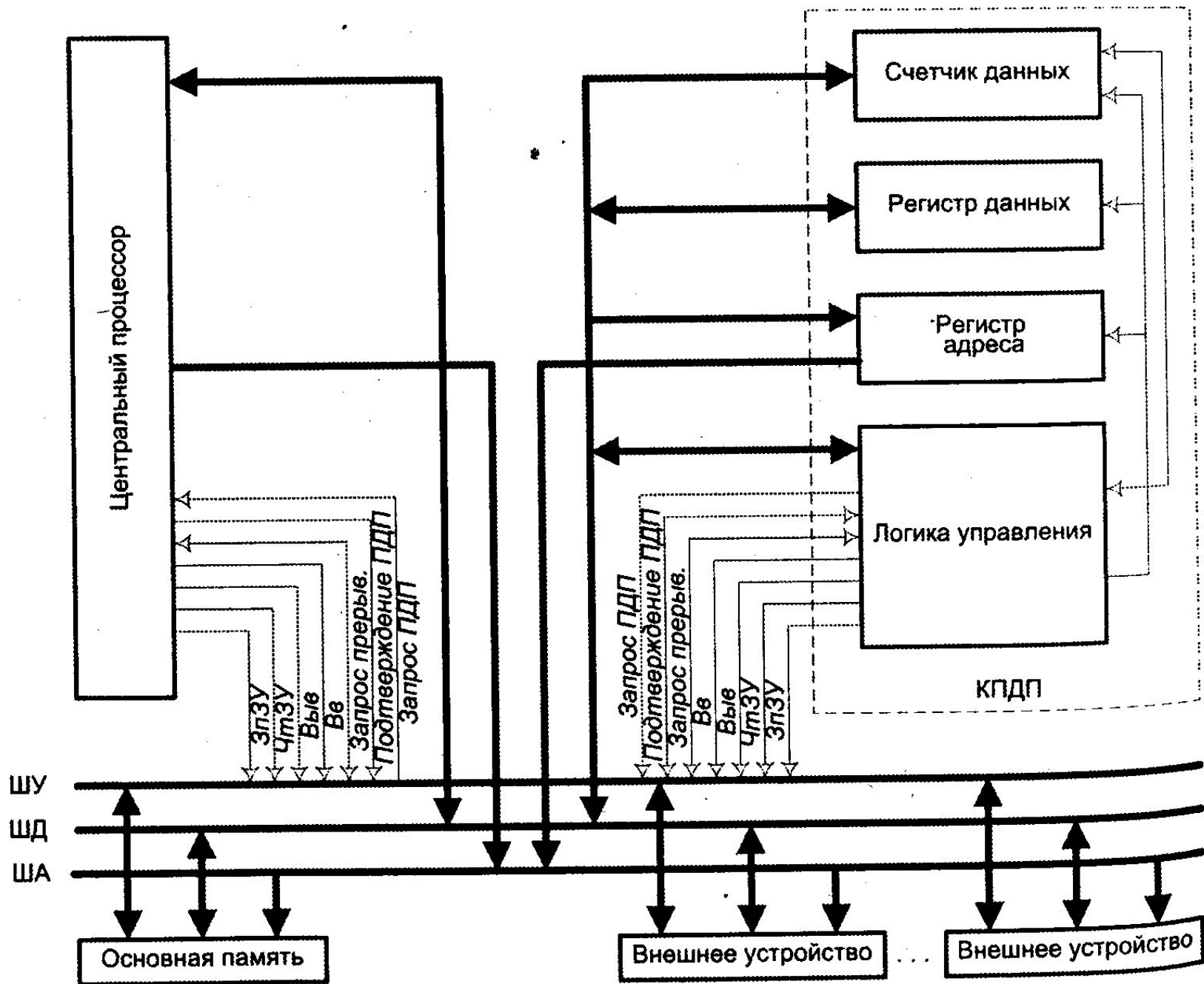


Рис. 8.9. Организация прямого доступа к памяти

Фрагмент процесса

После инициализации процесс пересылки информации может быть начат в любой момент.

Инициаторами обмена вправе выступать как ЦП, так и ВУ. Устройство, желающее начать В/ВЫВ, извещает об этом контроллер подачей соответствующего сигнала.

Получив такой сигнал, КПДП выдает в ЦП сигнал «*Запрос ПДП*». В ответ ЦП освобождает шины адреса и данных, а также те линии шины | управления, по которым передаются сигналы, управляющие операциями на шине адреса (ША) и шине данных (ШД). К таким, прежде всего, относятся линии *ЧтЗУ*, *ЗпЗУ*, *Выв*, *Вв* и линия выдачи адреса на *ША*.

Далее ЦП отвечает контроллеру сигналом «*Подтверждение ПДП*», который для последнего означает, что ему делегированы права на управление системной шиной и можно приступать к пересылке данных.

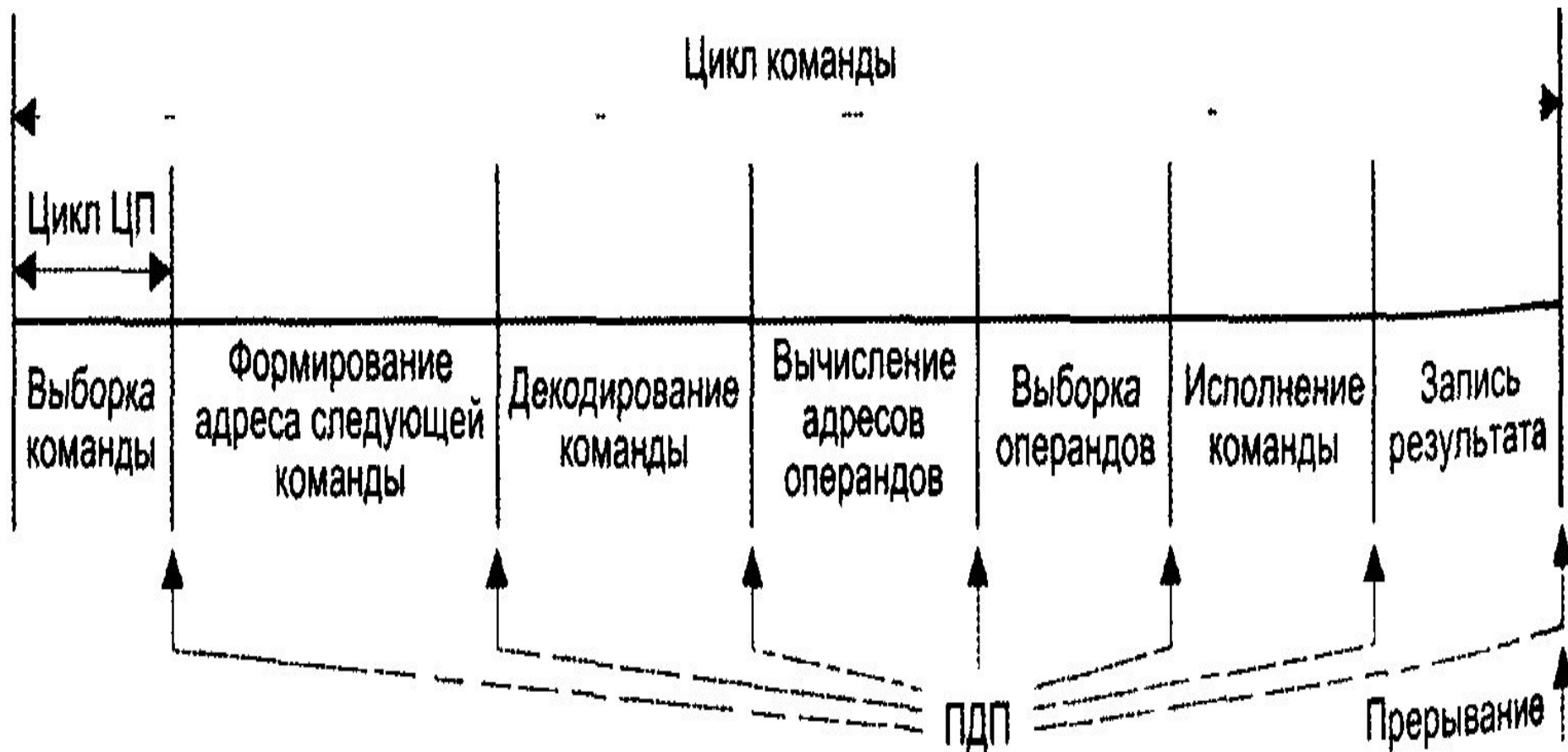
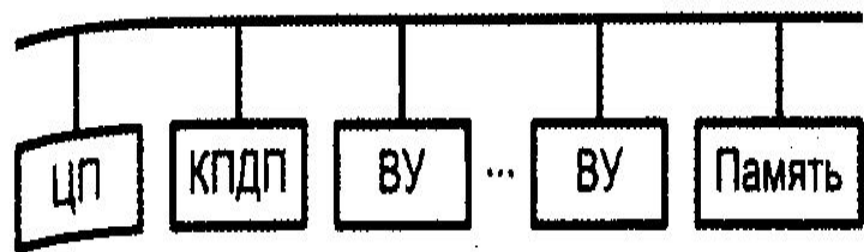
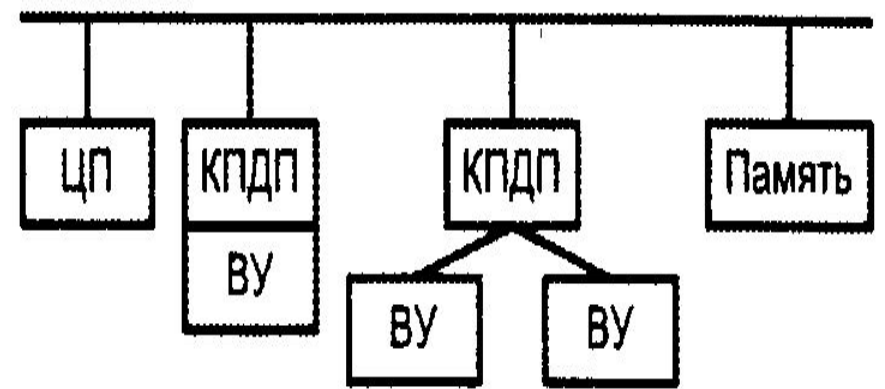


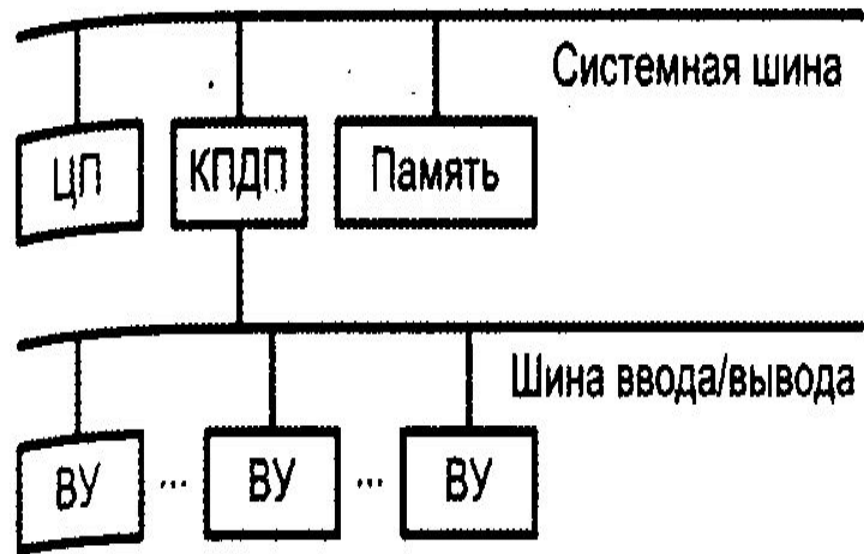
Рис. 8.10. Точки возможного вмешательства в цикл команды при прямом доступе к памяти и при обычном прерывании



а



б



в

Рис. 8.11. Возможные конфигурации систем прямого доступа к памяти

Каналы и процессоры ввода/вывода

По мере развития систем В/ВЫВ их функции усложняются. Главная цель такого усложнения — максимальное высвобождение ЦП от управления процессами ввода/вывода. Следующими шагами в преодолении проблемы могут быть:

1. Расширение возможностей МВВ и предоставление ему прав процессора со специализированным набором команд, ориентированных на операции ввода/вывода. ЦП дает указание такому процессору В/ВЫВ выполнить хранящуюся в памяти VM программу ввода/вывода. Процессор В/ВЫВ извлекает и исполняет команды этой программы без участия центрального процессора и прерывает ЦП только после завершения всей программы ввода/вывода.

2. Рассмотренному в пункте 1 процессору ввода/вывода придается собственная локальная память, при этом возможно управление множеством устройств В/ ВВВ с минимальным привлечением ЦП.

- В первом случае МВВ называют *каналом ввода/вывода* (КВВ), а во втором — *процессором ввода/вывода*. В принципе различие между каналом и процессором ввода/вывода достаточно условно, поэтому в дальнейшем будем пользоваться термином «канал».
- Концепция системы ввода/вывода с КВВ характерна для больших универсальных вычислительных машин (мэйнфреймов), где проблема эффективной организации В/ВВВ и максимального высвобождения центрального процессора в пользу его основной функции стоит наиболее остро. СВВ с каналами ввода/вывода была предложена и реализована в ВМ семейства IBM 360 и получила дальнейшее развитие в семействах IBM 370 и IBM 390.

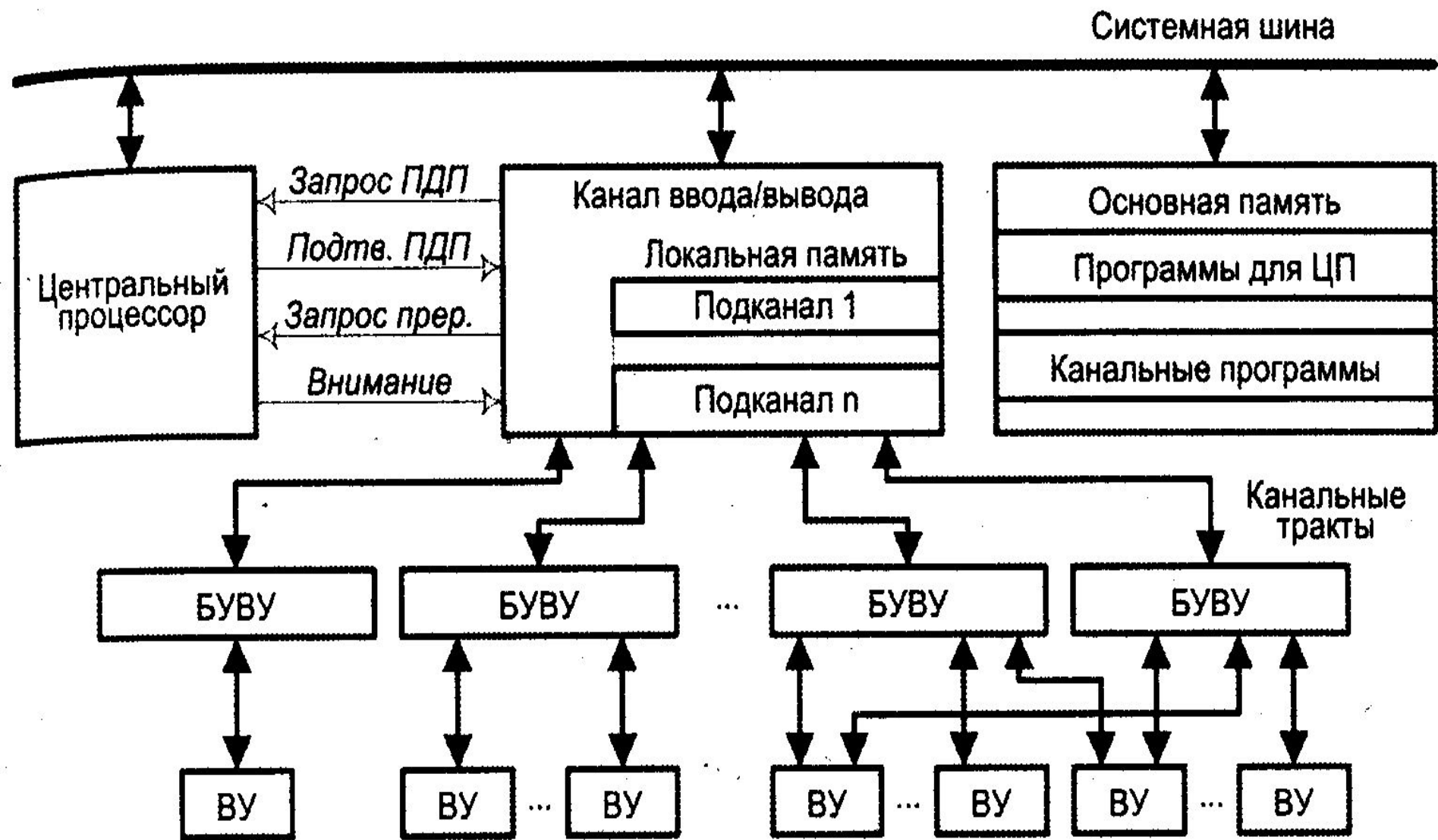


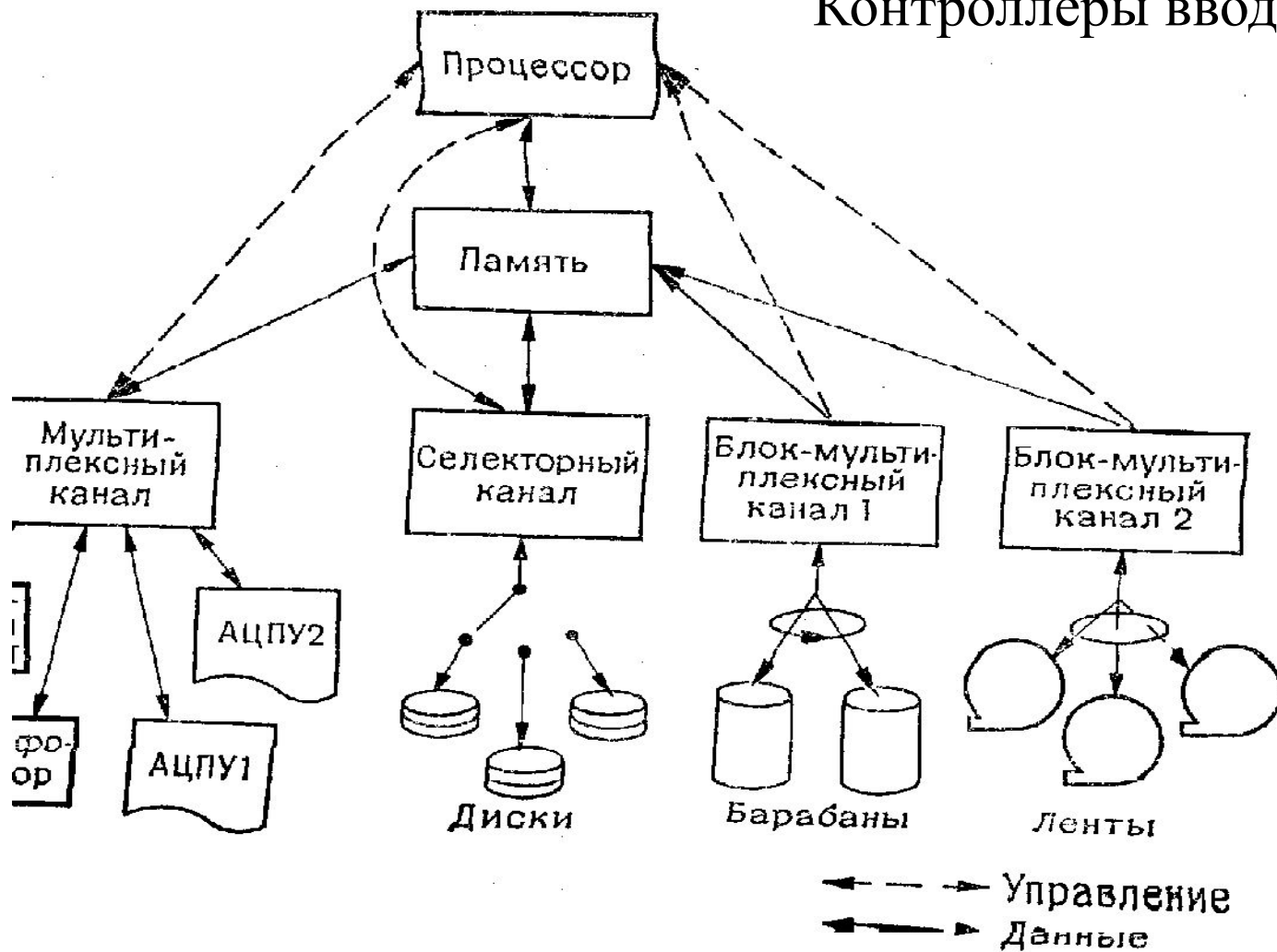
Рис. 8.12. VM с канальной системой ввода/вывода

Подсистема ввода-вывода.

1. Типы каналов ввода-вывода.

Основные параметры и принципы работы подсистемы ввода-вывода. Режимы работы каналов. Интерфейсы ввода-вывода.

Контроллеры ввода-вывода.



2. Концепции программирования ввода-вывода.

- Для пояснения проблем, которые встают перед операционной системой, введем некоторые основные понятия, относящиеся к программированию ввода-вывода.
- Центральный процессор обычно устанавливает связь с процессором ввода-вывода с помощью специальных команд, таких, как **НАЧАТЬ ВВОД-ВЫВОД** и **ОСТАНОВИТЬ ВВОД-ВЫВОД**. Процессор ввода-вывода может связаться с CPU с помощью прерываний.
- Процессор ввода-вывода имеет свою собственную систему команд. Чтобы различать команды CPU и команды канала ввода-вывода, мы будем называть последние *командами ввода-вывода*. Программы, написанные с помощью этой системы команд, называются **программами ввода-вывода**. Программисты обычно не пишут собственные программы ввода-вывода, а обращаются к системе, которая обеспечивает их программами ввода-вывода.

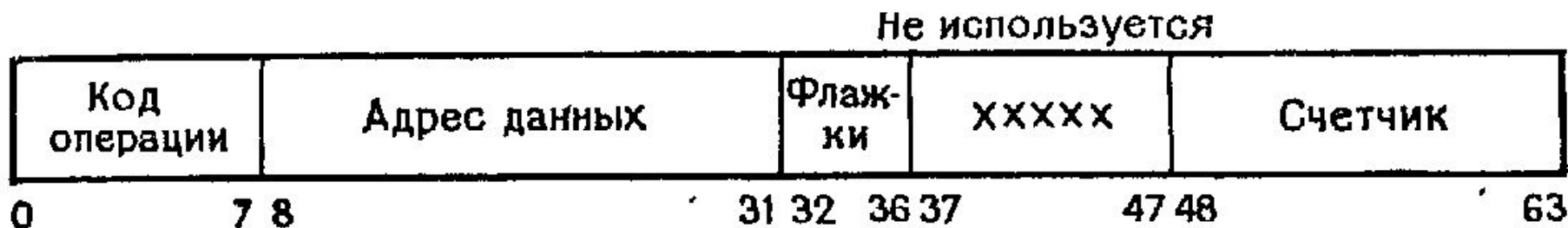
3. Структура процессора ввода-вывода Системы 370.

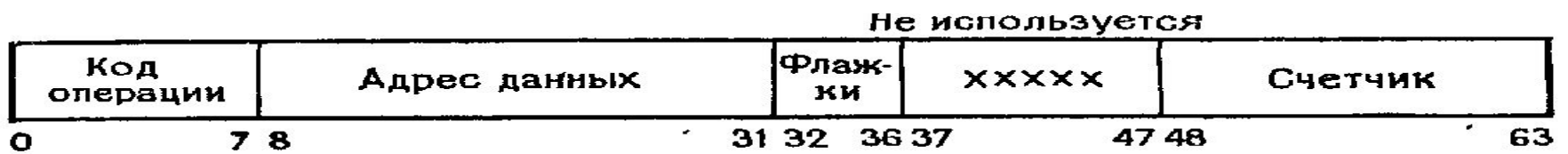
Процессоры ввода-вывода не достигли такого же уровня стандартизации команд, какой мы наблюдаем у CPU. Однако их основные функции поддаются обобщению.

- **3.1. Память.** Память та же самая, какая используется в центральных процессорах Системы 370; основная единица памяти - байт, емкость памяти - до 2^{24} байтов. Для адресации в каналах ввода-вывода используются 24-битовые абсолютные адреса.
- **3.2. Регистры.** Сколько имеется регистров? Каковы их размеры, функции и взаимосвязь? Канал ввода-вывода не имеет регистров в явном виде, но имеет регистр адреса команды и счетчика данных. Некоторые устройства ввода-вывода имеют внутренние регистры, подобные внутренним рабочим регистрам CPU, используемым при выполнении команд.
- **3.3. Данные.** Данные, каких типов могут обрабатываться процессором. Прежде всего, могут обрабатываться логические символьные данные (строка последовательных байтов длиной (2 в 16-ой байтов); байты уточненного состояния, характеризующие состояние устройств ввода-вывода, и может читать и обрабатывать данные этого типа.

3.4. Команды.

- Какие классы команд выполняются процессором Ввода-вывода? Имеется три основные группы команд ввода-вывода:
- 1. Команды передачи данных: читать; читать в обратном направлении; писать; уточнить состояние (читать состояние устройства).
- 2. Команды управления устройством: управление (пропуск страницы, перемотка ленты и т. д.).
- 3. Команды перехода: передача управления внутри канальной программы.
- Канал выбирает команды канала (Channel Command Word - CCW) из памяти и декодирует их в соответствии со следующим форматом:





Код операции (биты 0 - 7) определяет команду, которая должна быть выполнена; Адрес данных (биты 8 - 31) определяет адрес первого байта данных в команде.

Поле счетчика определяет длину в байтах поля данных..

Флажки служат для дальнейшей конкретизации данной команды:

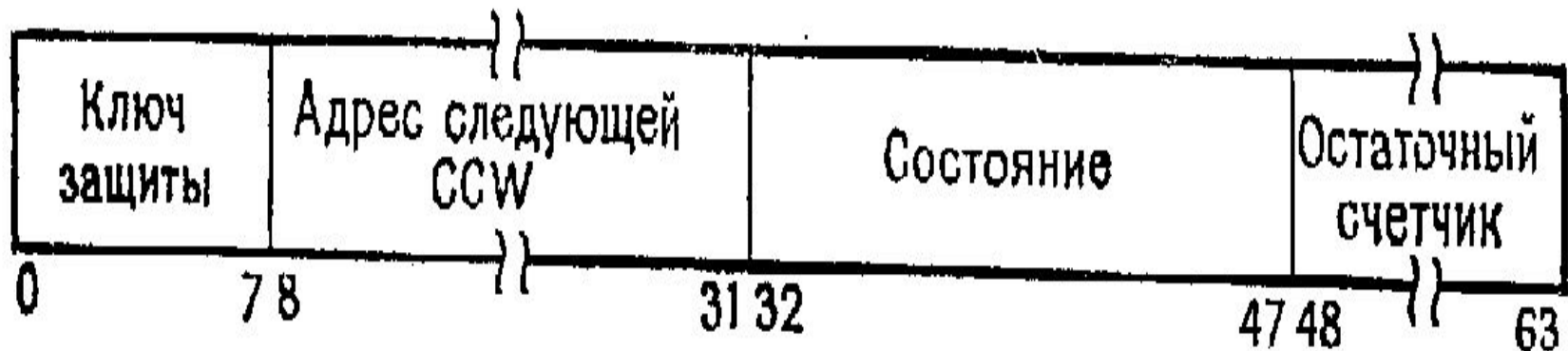
1. Флажок цепочки команд (бит 33) означает, что после завершения текущей команды должна выполняться следующая по порядку команда канала. (Если флажок = 0, то канал останавливается.)

2. Флажок цепочки данных (бит 32) означает, что текущая операция должна быть продолжена над областью данных, определенной в следующей ССW. Операция продолжается до обнуления счетчика данных.

- 3. Флажок подавления индикации неправильной длины (бит 34) блокирует индикацию, с помощью которой программе передается информация о несоответствии длины .
- 4 Флажок блокировки записи (бит 35) позволяет не допустить передачу информации в память.
- 5. Флажок программно управляемого прерывания (бит 36) вызывает формирование в канале условия прерывания, когда эта ССW начинает выполняться каналом.
- Несколько ССW, соединенных вместе с помощью флажков цепочки команд или цепочки данных либо командами перехода, *называются канальной программой*. CPU инициирует канал, чтобы он выполнял канальную программу.

3.5. Специальные средства.

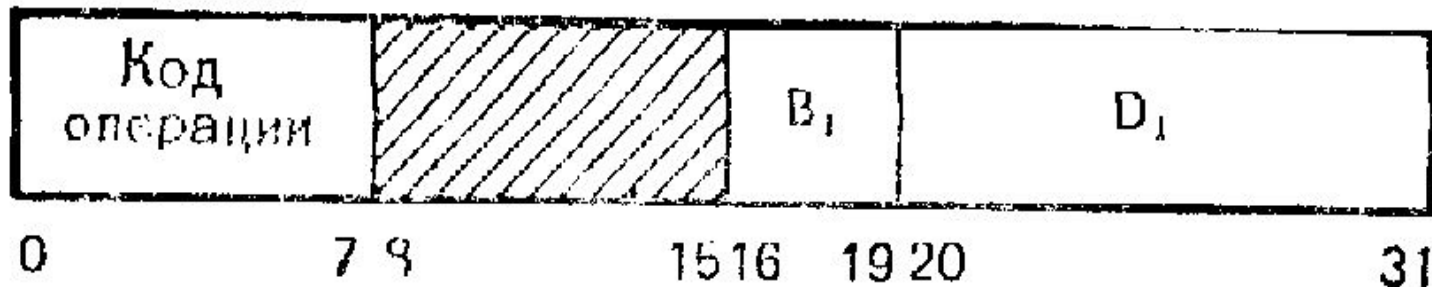
- *Состояния.* Канал имеет внутренний регистр, функции которого совпадают с функциями регистра адреса команд. (Мультиплексный или Блок-мультиплексный канал имеет несколько таких регистров, по одному на каждое устройство.) Кроме того, для хранения информации о состоянии в памяти зарезервированы три специальных слова. **Адресное слово канала** (Channel Address Word - CAW), расположенное в памяти в ячейке 72, содержит адрес первой команды, которую должен выполнить канал. Канал обращается к CAW только во время выполнения центральным процессором команды НАЧАТЬ ВВОД-ВЫВОД (SIO). **Слово состояния канала** (Channel Status Word - CSW), занимающее на самом деле поле длиной в двойное слово, содержит в закодированном виде информацию о состоянии канала. CSW расположено в памяти в ячейке 64 и имеет следующий формат:



Ключ - ключ защиты, используемый каналом.; **Адрес** - адрес следующей команды канала. **Состояние** - например, завершение операции ввода-вывода, возникновение ошибок ввода-вывода и т. д. **Счетчик** - сколько байтов не обработала последняя ССW (обычно счетчик равен 0, если только операция ввода-вывода не была аварийно прекращена)

5. Связь между CPU и каналом.

- До сих пор мы рассматривали работу канала. Теперь перейдем к рассмотрению связи канала и CPU. Основное назначение канала - освободить CPU от управления операциями ввода-вывода во всех деталях. CPU и канал обычно находятся в отношениях, подобных отношениям хозяина и слуги. Это означает, что CPU указывает каналу, когда начинать операцию, и приказывает ему остановиться или изменить выполняемую операцию. С другой стороны, канал обычно не начинает операции, пока не получит указания со стороны CPU.



- Между каналом и CPU существует двусторонняя связь:
- 1) от CPU к каналу - команды ввода-вывода процессора, инициируемые CPU;
- 2) от канала к CPU - прерывания, инициируемые каналом.
- В этом разделе мы рассмотрим первый тип связи - взаимоотношения между командами ввода-вывода процессора и каналом. Второй тип связи - через систему прерываний ввода-вывода.
- Все команды ввода-вывода CPU имеют следующий формат:

Три командами ввода-вывода CPU

- Номер канала, и номер устройства образуются путём сложения содержимого регистра В1 и содержимого поля D1. Биты 16 - 23 суммы формируют адрес канала, а биты 24-31 - адрес устройства, подключенного к каналу
- **В основном мы познакомимся с тремя командами ввода-вывода CPU:**
- 1. **НАЧАТЬ ВВОД-ВЫВОД (SIO).** Для того чтобы начать операцию ввода-вывода, необходимо знать 1) адрес канала и 2) начальный адрес канальной программы. Команда **НАЧАТЬ ВВОД-ВЫВОД**, например **SIO X'00E'**, определяет адрес канала - 0 и адрес устройства - 0E. В ячейках памяти 72-75 находится **CAW**, содержимое, которого определяет начальный адрес канальной программы.

- 2. ПРОВЕРИТЬ ВВОД-ВЫВОД (ТІО). CPU индицирует состояние адресуемого канала и устройства путем установки признака результата (занято или нет). Признак результата может быть, затем проверен с помощью стандартной команды условного перехода.
- 3. ОСТАНОВИТЬ ВВОД-ВЫВОД (НІО). Выполнение текущей операции на адресуемом канале и устройстве ввода-вывода мгновенно прекращается.
- После выполнения команд SIO или TIO CPU может получить следующий признак результата: 8 - доступен; 2 - занят; 1 - не работает; 4 - указывает на наличие дополнительной информации, сообщаемой с помощью CSW, которое находится в ячейке памяти 64.

Работа системы

- Слово состояния канала (CSW) содержит подробную информацию о состоянии устройства ввода-вывода или об условиях, которые вызвали прекращение операции ввода-вывода. Информация в CSW может быть записана во время прерывания ввода-вывода и иногда во время выполнения команд НАЧАТЬ ВВОД-ВЫВОД, ОСТАНОВИТЬ ВВОД-ВЫВОД и ПРОВЕРИТЬ ВВОД-ВЫВОД. Команда SIO инициирует выполнение операции ввода-вывода только в том случае, если канал на команду SIO ответил признаком результата, равны 8. Если получено любое другое знака результата, то это означает, что канал отверг запрос на ввод-вывод. Причины неприятия канал описываются в CAW или сообщаются с помощью результата.
- Хотя аппарат прерываний ввода-вывода обладает мощными возможностями, в простейших случаях обработки ввода-вывода его применение не является необходимым. Например, приведенная ниже последовательность команд инициирует выполнение программы вывода и проверяет ее завершение в предположении, что прерывания ввода-вывода замаскированы:

Пример.

⋮

LA 1,CSWADDR Поместить адрес программы ввода-вывода в CSW

ST 1,72

SIO X'00E' Начать операцию ввода-вывода на устройстве
с адресом X'00E'

BC 4+2+1,ERROR Если CS ≠ 8, то ошибка

STIO TIO X'00E' Проверить, не завершилась ли операция ?

BC 4+2,TESTIO Если нет, то повторить проверку

⋮

Если мы зафиксируем признак результата каждой команды
ПРОВЕРИТЬ ВВОД-ВЫВОД, то получим последовательность:

Признак результата ТЮ

2 - занят

2 - занят

2 - занят, операция ввода-вывода выполняется

4 - записано CSW операция ввода-вывода завершена

8 - доступен

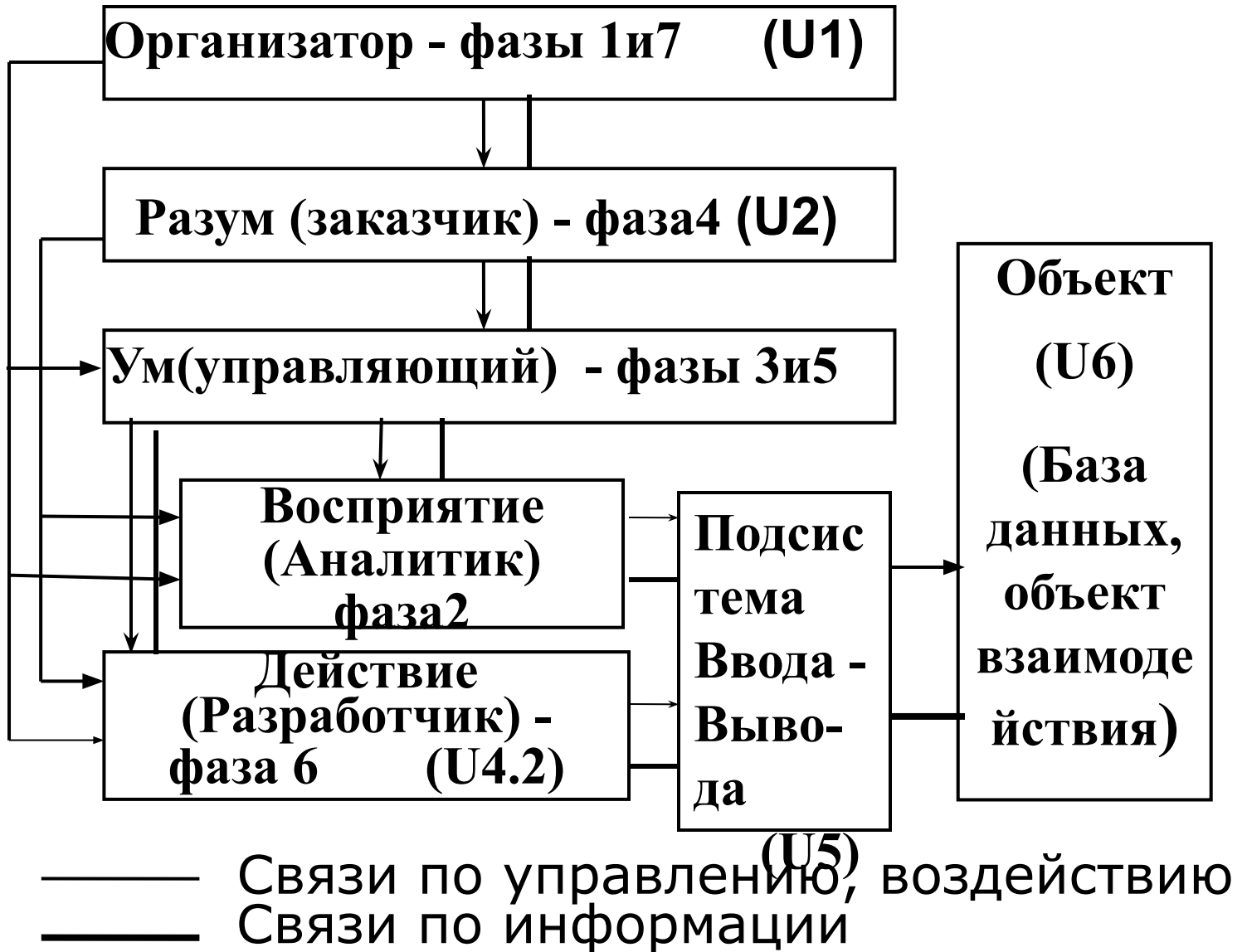
цикл завершен, CS не равно ни 4, ни 2.

Два режима: без и с прерываниями.

- Не исключено, что придется выполнить тысячи команд ТЮ, признак результата которых будет равен 2. Например, печать одной строки занимает приблизительно 60 мс (при скорости 1000 строк в минуту). Если команды ТЮ и УСЛОВНЫЙ ПЕРЕХОД (ВС) выполняются за 1 мкс, то эти команды будут повторены в цикле 30000 раз, причем каждый раз $CS = 2$ (занято). Это является одним из аргументов за применение механизма прерываний. Когда же операция ввода-вывода завершится, то возникнет условие *ждущего прерывания* - канал пытается сообщить CPU о том, что операция ввода-вывода завершились, однако прерывания ввода-вывода замаскированы. Признаком результата следующей команды уже не 2, поскольку устройство больше не занято выполнением операции ввода-вывода. Мы получим признак результата 4, это означает, что информация о состоянии (CSW) записана в ячейки 64 - 71.

- В этом случае программа могла бы проверить поле состояния в CSW и убедиться, что в самом деле операция ввода-вывода закончилась успешно. Поскольку канал передал CPU всю нужную информацию, он автоматически сбрасывает условие *ждущего прерывания*. Таким образом, следующая команда TIO найдет устройство не занятым, не обнаружит ждущих прерываний и установит признак результата 8, указывающий, что операция завершилась.

Структурная схема учебной деловой игры.



Игра

Любой процесс взаимодействия между субъектом и объектом по разрешению ПС в составе УДИ и её участников содержит семь фаз своей реализации:

1. Наличие совокупности предпосылок W или существование основы процесса (Для участника это Первопричина, личность обучаемого как целое);
2. Восприятие (ощущение) текущего состояния S и внутренней потребности P из памяти (Для участника это Чувства);
3. Сравнение состояния S с потребностью P и формирование отклонения желаемого от действительного (Для участника это Ум);
4. Постановка задачи, различение, выбор направления устранения противоречия между текущим состоянием и потребностью (Для участника это Разум);
5. Управление локальными целями, или желаниями, устранением отклонения через воздействия на основу S и P (Для участника это Ум);
6. Использование сил и средств трансформации процесса (Объект/Субъект) (Для участника это Органы воздействия);
7. Реализация (поглощение) полученных на этапах 2-6 результатов процесса W , созревание и проявление обновлённой *причины* на первом уровне взаимодействия.

КОНТРОЛЛЕР ПРЯМОГО ДОСТУПА К ПАМЯТИ

- Прямой доступа к памяти (Direct Memory Access - DMA) применяется для выполнения операций передачи данных непосредственно между оперативной памятью и устройствами ввода/вывода. Обычно это такие устройства, как НГМД, НМД, стримеры.
- При использовании DMA процессор не участвует в операциях ввода/вывода. Контроллер прямого доступа сам формирует все сигналы, необходимые для обмена данными с устройством. Скорость такого непосредственного обмена значительно выше, чем при традиционном обмене с использованием центрального процессора и команд INP, OUT.
- Заметим, что контроллеры DMA в компьютерах IBM PC/XT и IBM PC/AT различаются, но совместимы снизу вверх. Поэтому вначале мы рассмотрим о первый тип контроллеров, затем займемся контроллером DMA компьютера IBM PC/AT.
- Контроллер

Контроллер прямого доступа IBM PC/XT

- Контроллер прямого доступа IBM PC/XT реализован на базе микросхемы Intel 3237A и содержит 4 канала. Эти каналы используются следующим образом:

Номер канала. Для чего применяется.

- | | |
|---|---|
| 0 | Обновление содержимого динамической памяти компьютера. Этот канал имеет наивысший приоритет |
| 1 | Не используется |
| 2 | Адаптер накопителя на гибком магнитном диске (НГМД) |
| 3 | Адаптер накопителя на магнитном диске (НМД). Этот канал имеет низший приоритет |

Каждый канал DMA содержит следующие 16-разрядные регистры:

- регистр текущего адреса CAR, содержит текущий адрес ячейки памяти при выполнении операции обмена данными с использованием DMA;
- регистр циклов прямого доступа к памяти CWR; содержит число слов предназначенных для передачи, минус единица. При выполнении обмена данными регистр работает в режиме вычитания;
- регистр хранения базового адреса BAR; используется для хранения адреса памяти, применяемого при передаче данных. В процессе работы канала DMA содержимое этого регистра не изменяется;
- регистр хранения базового числа циклов прямого доступа к памяти WCR; хранит число циклов DMA, и его содержимое также не изменяется;
- регистр режима MR, определяющий режим работы канала.

MR имеет следующие поля:

- | Поле | Описание |
|-------------|---|
| • 0-1 | Номер канала (00 - 0; 01 - 1; 10 - 2; 11 - 3) |
| • 2-3 | Тип цикла DMA: 00 - цикл проверки; 01 - цикл записи; 10 - цикл чтения; 11 - запрещённая комбинация |
| • 4 | 1 - режим автоинициализации |
| • 5 | Приращение адреса: 0 - инкрементированное
1 - дискрементированное |
| • 6-7 | Режимы обслуживания: 00 - передача по требованию;
01 - одиночная передача; 10 - передача по блокам;
11 - каскадирование |

Инициализация канала DMA.

- Для инициализации канала программа должна выполнять следующие шаги:
- ◆ сбросить триггер байтов командной записи в регистр 0Ch;
- ◆ задать режим работы канала, выполнив запись по адресу 0Vh в регистр режима MR;
- ◆ записать младшие 16 бит 20-битового адреса области памяти, которая будет использоваться для передачи данных, в регистр базового адреса. Адрес порта зависит от номера канала: канал 0 использует адрес 00h, канал 2 - 04h, канал 3 - 06h;
- ◆ записать номер страницы (старшие 4 бита 20-битового адреса) в регистр страниц 81h;
- ◆ загрузить регистр циклов прямого доступа к памяти CWR значением, на единицу меньшим требуемого количества передаваемых байт. Адреса соответствующих портов для каналов 0-3, равны соответственно 01h, 031h, 051h и 07h;
- ◆ разрешить работу канала, выполнив запись в регистр маски каналов по адресу 0Ah.
- Сразу после разрешения канал начинает передачу данных. После окончания передачи устройство обычно вырабатывает прерывание, которое служит признаком окончания операции ввода или вывода данных.

Ещё один вариант объяснения принципа работы DMA

Ввод-вывод с прямым доступом к памяти предназначен для внешних устройств, у которых передача данных осуществляется с большой скоростью. Так, в накопителях на гибком магнитном диске скорость передачи данных составляет 31,25 Кбайт/с при одинарной плотности записи и 62,5 Кбайт/с—при двойной плотности. При таких скоростях передачи данных между ОП и ВУ производительность (пропускная способность) МП в режиме прерывания недостаточна. Для МП КР580 программа обслуживания

ВУ выполняется за 50 мкс, в то время как интервал времени между байтами у накопителя на гибких магнитных дисках составляет 32 мкс для одинарной плотности записи и 16 мкс для двойной плотности.

Для обеспечения необходимой скорости передачи данных между ОП и ВУ разработан метод прямого доступа к памяти (ПДП), благодаря которому передача данных происходит непосредственно (напрямую) между ОП и ВУ без участия МП. Для этих целей используются специальные аппаратные средства (контроллеры ПДП), которые выполняют функции управления передачей данных между ОП и

Структура контроллера

ВУ.

В режиме прямого доступа происходит временное разделение (мультиплексирование) системной магистрали между МП и контроллером ПДП. В обычных режимах управление системной магистралью возлагается на МП. При инициализации режима прямого доступа МП отключается от системной магистрали и передает управление контроллеру ПДП, который с необходимой скоростью обеспечивает передачу данных между ОП и ВУ.

Для того чтобы контроллер ПДП мог осуществлять управление передачей данных, он должен в свой состав включить электронные схемы, которые задают режим передачи данных ОП-ВУ или ВУ-ОП; формируют адреса в ОП для записи или чтения данных; определяют границы передачи данных в ОП (объем передаваемых данных); реагируют на запросы ВУ в момент его готовности для передачи очередного кода символа. Обеспечение перечисленных функций в контроллере осуществляют регистры управления и состояния, счетчик слов и указатель буфера (рис. 4).

Структурная схема контроллера

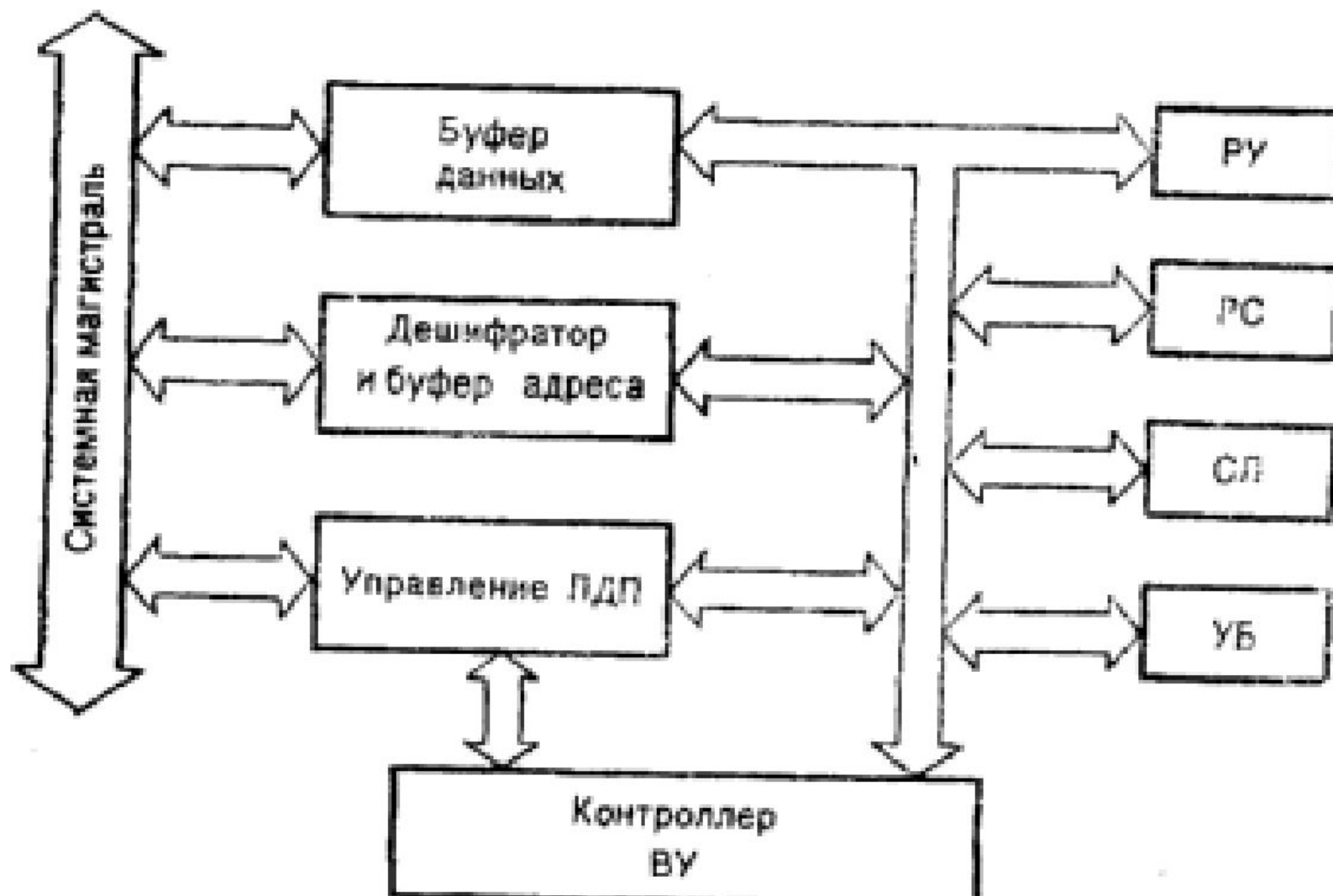


Рис. 4. Структурная схема контроллера ЛДП

Назначение регистров контроллера DMA

Регистр управления (РУ) запоминает команды от МП, которые включают разрешение режима прямого доступа к памяти, тип передачи для ОП (чтение или запись), состояние прерывания (разрешены или запрещены).

Регистр состояния (РС) запоминает информацию о состоянии контроллера ПДП. По отношению к МП контроллер ПДП выступает как ВУ, поэтому для организации функционирования ПЭВМ и режиме ПДП МП должен иметь возможность получения информации о состоянии контроллера ПДП: готовность к управлению передачей данных, сбойные ситуации во время передачи данных и т. д.

Счетчик слов (СЛ) используется для загрузки длины передаваемого блока. После передачи очередного слова между ОП и ВУ содержимое СЛ уменьшается на единицу. Передача завершается, когда содержимое счетчика достигает нуля.

Указатель буфера (УБ) запоминает текущий адрес ячейки ОП, к которой производится обращение при передаче очередного слова. Перед началом передачи

в УБ загружается начальный адрес блока данных в ОП. В процессе передачи данных осуществляется автоматический инкремент УБ, что позволяет передать блок данных в последовательные ячейки ОП.

Описание принципа обмена данными.

Обмен данными между ВУ и ОП в режиме ПДП предусматривает выполнение некоторой последовательности операций, включающих:

инициализацию передачи блока данных между ОП и ВУ, которая осуществляется в результате генерации со стороны контроллера ВУ сигнала запроса ПДП;

формирование контроллером ПДП сигнала запроса прерывания (ЗПР), на который МП реагирует сигналом подтверждения прерывания (ППР);

передачу из МП в регистры контроллера ПДП (РУ, СЛ, УБ) управляющей информации, необходимой для обмена данными между ОП и ВУ;

генерацию контроллером ПДП сигнала захвата (ЗХВ), по которому МП отключается от системной магистрали, а контроллер ПДП переходит в режим управления системной магистралью.

При передаче блока данных между ОП и ВУ контроллер ПДП генерирует сигналы чтения данных из ВУ и записи данных в ОП (ЧТВУ—ЗПОП) или записи данных во ВУ и чтения данных из ОП (ЗПВУ—ЧТОП), а также текущие адреса данных в ОП (содержимое УБ).

Завершение цикла передачи блока данных между ОП и ВУ включает снятие сигнала ЗХВ и генерацию сигнала ЗПР. По сигналу ЗПР МП переходит на обслуживание контроллера ПДП: читает и анализирует содержимое РС, переходит к выполнению основной программы при нормальном завершении цикла передачи блока данных, передаст управление программе обработки сбойных ситуаций при регистрации ошибок.

Как вписывается в эту картину духовный аспект?

(В129) Вопрос: Мы повсюду слышим о сотовых телефонах и дистанционном управлении. Похоже, все нужды нашей земной жизни скоро будут обеспечиваться электроникой и компьютерами, которые привлекают всеобщее внимание. Как вписывается в эту картину духовный аспект? Остается ли для него время? Вдобавок к этому, мы слышим об испытаниях ядерного оружия. Как нам сочетать и объединять эти столь разные сферы? Пожалуйста, укажи нам путь.

Ответ: Люди ценнее всех богатств мира. Если уж вы родились людьми, вам нужно стремиться достичь освобождения.

Место компьютера

(В67) Вопрос: Сейчас, говоря со студентами и спрашивая их, чем они заняты, мы часто слышим, что они «сидят в интернете» или осваивают очередную компьютерную программу. Мы хотим знать Ваше мнение по поводу этой компьютерной мании.

Ответ: Одна из главных вещей, которую вам следует знать, - это то, что все ваше современное образование находится внутри вас. Вы сами - «компьютер». Вы - «генератор». Внутри вас - «электричество». Внутри вас - «тепловая энергия». Вы - «магнит». Все, что вас окружает, - не что иное, как «отражение», «реакция», «отзвук» вашего внутреннего существа.

Кто моя сестра?

Вы говорите, что компьютер - великое изобретение. Встаньте перед компьютером и спросите: «Кто моя сестра?» Получите ли вы ответ? Нет.

Почему? Компьютер нужно начинить определенной информацией, которую вы потом, когда вам нужно, извлекаете из него, не так ли? Он «выдает» лишь то, что уже прежде было в него заложено. Ничего нового! В конце концов, кто создал компьютер? Человек. Поэтому, естественно, человек гораздо более велик, чем компьютер. Ваш мозг - это компьютер, где запечатлены весь ваш опыт и все воспоминания. Даже если во время беседы со Мной вас кусает комар, вы смахнете его. Каким образом? Это внутренние связи в вас. Когда на дороге вы замечаете колючку, нога автоматически ступает в сторону, она не наступит на колючку. Почему? Это магнитная сила, действующая в вас.

Кто - мастер?

Когда Вас спрашивают вас: «Чем вы занимаетесь?», некоторые отвечают: «Учусь на магистра, технических наук» или «Я - магистр (master) технических наук». Но кто - мастер? Что такое «техника»? Мастер, хозяин - тот, кто умеет контролировать свои чувства, и никто другой. Вы же просто «мистеры», а не «мастера». Вы должны использовать свое знание для того, чтобы общество совершенствовалось, а иначе это не знание, а головная боль (not knowledge, but allergy). Вам нужно идти в общество со своими знаниями. «Возьми с собой знание» - это и есть «технология». В противном случае это «трюкология». Она никогда не превратится в «технологию».

Наука -ниже уровня чувств.

Духовность - над чувствами.

Наука и техника могут сделать жизнь более удобной и комфортной. Но они не даруют вам **покоя и блаженства**. **Что такое наука? Что такое духовность? Какая между ними разница? Наука -ниже уровня чувств.**

Духовность - над чувствами. Ты есть То. Духовность имеет дело с «Тем», а наука говорит о «этом», как будто бы пытаясь научить вас, что «ты есть То». «Это», - природа, мир. «То» - Единый или Бог.

Духовность подобий букве «О», полной и законченной, а наука подобна букве «С», которая неполна, и пустое пространство внутри нее, начинаясь в одной точке, кончается где угодно. Духовность начинается там, где кончается наука, она ведет вас от мира к Богу.

Духовность позволяет вам ощущать и переживать Божественное в этом мире. Таков путь к объединению науки и духовности. **Духовность - невидимый принцип, стоящий за всем внешним - новинками науки и техническими приспособлениями.**

Наука занята инертным, а духовность — сознанием.

(В130) Вопрос: Противоречат ли друг другу наука и духовность? Мы жаждем узнать мнение по этому вопросу.

Ответ: Наука действует ниже уровня чувств, в то время как область действия духовности - над чувствами. Наука имеет дело с «этим», а духовность обращена к Тому, то есть к Божественному. Наука относится к тому, что «здесь», к ближайшему, а духовность сосредоточена на «далеком», запредельном. Наука преследует материальные цели, духовность ведет к высшей конечной цели.

Наука занята инертным, а духовность — сознанием. Духовность стремится к фундаментальной основе, а наука экспериментирует с надстройкой.

Наука- коллекция фактов. Духовность метафизична.

Наука - это раскол любви (split of love), духовность - сущность любви (spirit of love). Наука следует по внешнему пути , духовность - это внутренний интуитивный путь. Наука может увидеть очевидное, воспринимаемое, духовность углубляется в скрытое. Вы можете освоить науку с помощью обычных глаз, но духовное знание нуждается в глазах мудрости. Наука-коллекция фактов, которые внедряются в вашу голову. Духовность метафизична, она затрагивает сердце. Наука неполна, как буква «С», начинающаяся в одной точке и заканчивающаяся где угодно. Духовность - полна, всеобъемлюща, как буква «О».

Наука картина.

Но духовность -это рентгеновская пленка.

Наука поставляет средства, инструменты. Духовность учит, как использовать их. Например, нож, которым вы можете резать фрукты, овощи и т. д., можно использовать и для того, чтобы перерезать горло. Наука показывает то, что вы видите, - как фотография, картина. Но духовность - это рентгеновская пленка. Наука - «негативна», духовность - «позитивна».

Творение «негативно», а Творец - «позитивен». Заполните свое сердце «позитивной» духовностью, как заполняете водой резервуар. Органы чувств - краны, из которых вы получите воду любви. Наука устанавливает факты, но духовность символизирует неизменную истину.