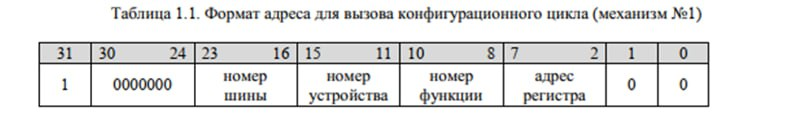
**Структура запроса для получения данных об устройстве PCI**



**Лабораторная работа №2 – Конфигурационное пространство PCI**

**1. Определение шины PCI. Отличия от шины PCI-Х, PCI-Express.**

***PCI*** – базовая системная шина компьютера архитектуры x86, предназначенная для подключения внутренних периферийных устройств и контроллеров внешних интерфейсов;

***PCI-X*** – шина для подключения рабочих станций и серверов, разработана на основе шины PCI;

Основная цель создания – улучшить ключевые характеристики шины PCI, прежде всего пропускную способность и надежность, за счет усложнения протокола обмена данными и увеличения тактовой частоты. Полная совместимость с устройствами PCI. На сегодня практически полностью вытеснена шиной PCI Express.

PCI-X пересмотрел традиционный стандарт PCI, удвоив максимальную тактовую частоту и объём данных, которыми обмениваются процессор компьютера и периферийные устройства. PCI-X также повышает отказоустойчивость PCI, позволяя повторно инициализировать или отключать неисправные карты.

PCI-X обратно совместим с PCI в том смысле, что вся шина возвращается к PCI, если какая-либо карта на шине не поддерживает PCI-X.

***PCI-Express*** – шина для подключения внутренних периферийных устройств и контроллеров внешних интерфейсов, разработанная на основе шины PCI.

В отличие от PCI протокол PCI Express условно разделен на уровни, без уточнения способов их реализации. Уровней всего 3, на каждом выполняется сборка и разборка пакетов и их обрамление необходимыми заголовками и контрольными суммами. Не все пакеты относятся к уровню транзакций, существуют пакеты только канального уровня, служащие для управления.

Основное различие между PCI и PCI Express заключается в том, что PCI является параллельным интерфейсом, а PCI Express - последовательным интерфейсом.

Также в отличие от стандарта PCI, использовавшего для передачи данных общую шину с подключением параллельно нескольких устройств, PCI Express, в общем случае, является пакетной сетью с топологией типа звезда.

**2. Какие новые технологии и интерфейсы конкурируют с PCI в современных компьютерах?**

1. ***PCI Express***

Хотя это не новая технология, PCIe продолжает развиваться с новыми стандартами, предлагая высокую пропускную способность и низкую задержку.

1. ***Thunderbolt***

Thunderbolt представляет собой высокоскоростной интерфейс для подключения периферийных устройств. Он использует разъем USB-C и поддерживает передачу данных, видео и питания через один кабель. Thunderbolt 4 также совместим с USB4.

1. ***NVMe (Non-Volatile Memory Express)***

Это протокол, который оптимизирован для работы с SSD, обеспечивая более высокую производительность по сравнению с традиционными SATA и SAS. NVMe снижает задержки и увеличивает пропускную способность.

1. ***USB 3.x и USB4***

USB 3.0 и новее (включая USB4) предлагают высокоскоростные соединения для подключения множества устройств, включая внешние накопители и периферийные устройства.

**3. Какие устройства могут быть подключены к шине PCI, и какие устройства являются типичными PCI-устройствами?**

Для подключения *внутренних периферийных устройств и контроллеров внешних интерфейсов*

Платы расширения: для добавления дополнительных функций, таких как параллельные или последовательные порты.

***Видеокарты*** – для обработки графики и вывода изображения на экран. Старые модели видеокарт могут использовать PCI.

***Сетевые карты*** — для подключения компьютера к локальным сетям или интернету как проводным, так и беспроводным способом.

***Звуковые карты*** — улучшают аудиовыход и позволяют подключать аудиосистемы, обеспечивают высококачественный звук и поддержку многоканального аудио.

**Контроллеры встраиваемых систем** — для подключения различных периферийных устройств, таких как принтеры или сканеры.

***Модемы*** — для подключения к интернету по телефонной линии.

***Контроллеры хранения данных*** — для подключения жестких дисков и SSD.

***USB-контроллеры*** — обеспечивают дополнительные порты USB для подключения периферийных устройств.

**4. Какие виды операций можно выполнять с устройствами, подключенными к шине PCI?**

***Операция чтения***

***Операция записи***

**5. Доступ к шине PCI и фазы транзакции.**

*Доступ к шине PCI:*

* До начала транзакции устройство инициатор подает запрос на доступ к шине
* Арбитр анализирует запросы и одному из устройств выдает разрешение
* Устройство, получившее разрешение, ожидает окончания текущей транзакции, после чего начинает работу.

*Фаза адресации*:

Устройство-инициатор обращается к целевому устройству с помощью адреса.

Фазу адресации отслеживают все устройства на шине, включая мост. То устройство, которое определило принадлежность адреса к своим ресурсам, сообщает об этом. Мост также может взять на себя роль целевого устройства, чтобы передать транзакцию на другие шины.

Если за 3 такта сигнала DEVSEL# не будет – произошла аварийная ситуация, и устройство заканчивает транзакцию с уведомлением своего драйвера.

*Фаза данных*:

Получив сигнал, инициатор готовит внутренние буферы к обмену и выставляет IRDY# по готовности. При выполнении записи в следующем такте на A/D поступает первая группа данных.

* Целевое устройство по готовности выставляет сигнал TRDY# и выставляет первую группу данных при выполнении чтения. Его отсутствие означает необходимость холостого такта, в течение которого инициатор прекращает передачу (или повторяет первую фазу данных).
* Холостой такт может вводить и инициатор снятием сигнала IRDY#.
* Маска байтов (линии C/BE#) определяет, какие байты из группы (по шине передаются сразу 4 или 8) заполнены полезными данными. Маску формирует тот, кто выставляет данные (при записи – инициатор, при чтении – целевое устройство).

*Завершение транзакции*:

Длина транзакции заранее неизвестна, поскольку она может быть закончена по инициативе трех агентов – инициатора, целевого устройства или арбитра.

Со стороны инициатора – снятием сигнала FRAME#:

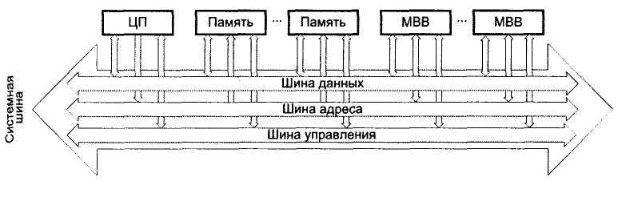
Со стороны целевого устройства – сигналом STOP#:

Со стороны арбитра – снятием сигнала GNT#.

**6. Схема подключения ПУ к системной шине.**

*Системная шина* служит для физического и логического объединения всех устройств ВМ, может иметь общую шину для памяти и устройств ввода-вывода.





Если один из модулей хочет передать данные в другой, он должен выполнить два действия: получить в свое распоряжение шину и передать по ней данные. Если какой-то модуль хочет получить данные от другого модуля, он должен получить доступ к шине и с помощью соответствующих линий управления и адреса передать в другой модуль запрос. Далее он должен ожидать, пока модуль, получивший запрос, пошлет данные.

**7. Классификация ПУ.**

Все устройства, не входящие в вычислительное ядро, относятся к ***периферийным***. Они могут располагаться снаружи / внутри корпуса ЭВМ, а также входить в состав основных микросхем системы.

***Основная задача периферийных устройств*** – поставка данных на обработку, а также вывод их за пределы вычислительного ядра.

***Деление ПУ по назначению***.

* Устройства ввода (Клавиатура, мышь)
* Устройства вывода (Дисплей, принтер)
* Устройства хранения данных (Винчестеры, дисководы)
* Сетевые и коммуникационные устройства (Модемы, сетевые адаптеры)

***ПУ по конструктивному исполнению***:

* Внешние;
* Внутренние;
* Встроенные.

**8. Методы управления вводом/выводом.**

В ВС находят применение три способа организации ввода/вывода:

1. ***Программно-управляемый ввод/вывод (******PIO);***

При PIO все связанные с этим действия происходят по инициативе процессора и под его полным контролем.

Выдав в контроллер ПУ команду, процессор должен ожидать завершения ее выполнения, и, поскольку он работает быстрее, чем ПУ, это приводит к потере времени.

1. ***Ввод/вывод по прерываниям;***

Во многом совпадает с PIO методом. Отличие состоит в том, что после выдачи команды ввода-вывода процессор не должен циклически опрашивать контроллер ПУ для выяснения состояния устройства.

Вместо этого процессор может продолжать выполнение других команд до тех пор, пока не получит запрос прерывания от контроллера ПУ, извещающий о завершении выполнения ранее выданной команды ввода-вывода. Как и при PIO, процессор отвечает за извлечение данных из оперативной памяти (при выводе) и запись данных в оперативную память (при вводе).

1. ***Прямой доступ к памяти.***

Повышение, как скорости ввода-вывода, так и эффективности использования процессора, обеспечивает третий способ ввода-вывода — ***прямой доступ к оперативной памяти (ПДП)***. В этом режиме оперативная память и контроллер ПУ обмениваются информацией напрямую.

**9. Характеристики шины PCI.**

Состоит из мультиплексированных линий передачи адреса и данных (разделение по времени) и линий различных управляющих сигналов.

* Разрядность (ширина):
  + 32 бита
  + 64 бита.
* Тактовая частота:
  + 33.3 МГц
  + 66.6 МГц.
* Адресация: (не зависит от ширины шины)
  + 32 бита
  + 64 бита
* Пропускная способность:
  + 132 Мбайт/с при 32-битной/33 МГц;
  + 264 Мбайт/с при 32-битной/66 МГц;
  + 264 Мбайт/с при 64-битной/33 МГц;
  + 528 Мбайт/с при 64-битной/66 МГц.
* Количество подключаемых устройств:
  + до 32 для одного физического сегмента шины, в зависимости от реализации.

**10. Основные принципы программирования доступа к периферийным устройствам**

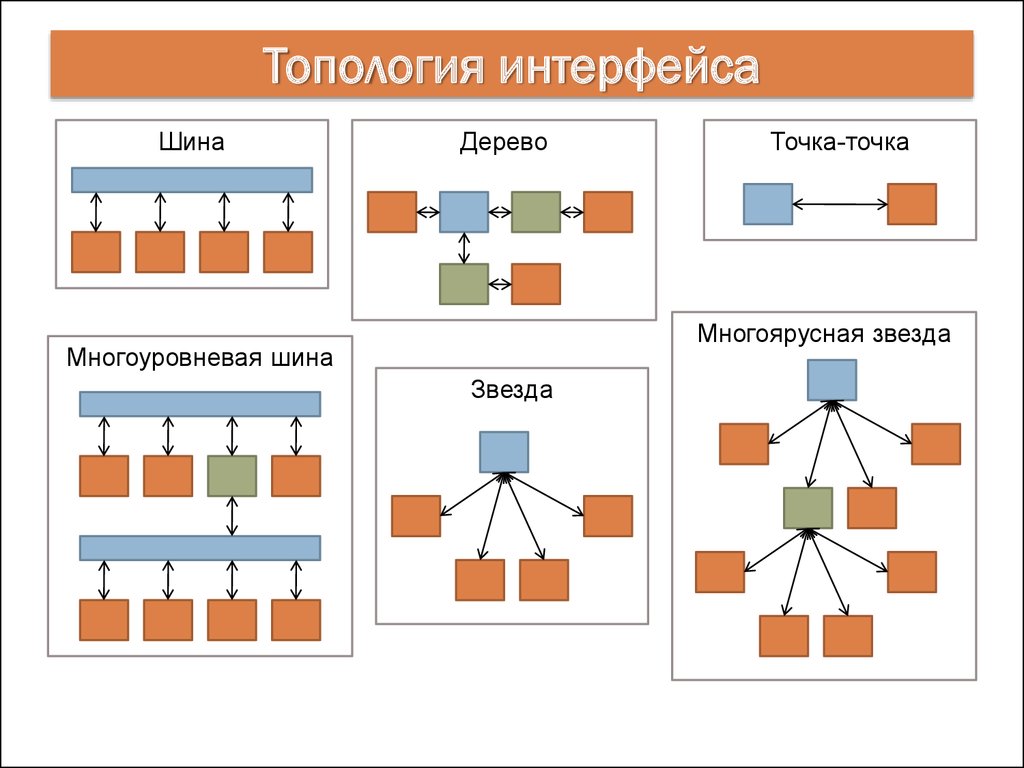
Разбиение ввода-вывода на несколько уровней, причем ***нижние*** – учитывают особенности аппаратуры, ***верхние*** обеспечивают удобный интерфейс для пользователей.

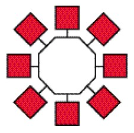
* независимость от устройств;
* обработка ошибок как можно ближе к аппаратуре;
* использование блокирующих (синхронных) и неблокирующих (асинхронных) передач;
* одни устройства являются разделяемыми, а другие выделенными.

Для решения поставленных проблем целесообразно разделить программное обеспечение ввода-вывода на четыре слоя:

1. Пользовательский слой ПО.
2. Независимый от устройств слой ОС.
3. Драйверы устройств.
4. Обработка прерываний

**11. Топологии интерфейса**





Кольцо

**12. Режимы кэширования шины PCI.**

***Режимы кэширования***:

**POSTED WRITE** — данные записи немедленно принимаются контроллером (мостом), который сразу же отправляет ответ «сделано» обратно устройству, инициировавшему запись. После этого мост пытается выполнить операцию записи на ведомой шине, но не блокирует дальнейшие операции.

**WRITE COMBINING** — оптимизирует несколько последовательных записей, которые идут по адресам, объединяя их в одну «взрывную» транзакцию. Это означает, что вместо отправки каждой записи по отдельности, мост собирает их и отправляет в одном пакете на ведомую шину.

**PREFETCHING** — Prefetching используется при транзакциях чтения и позволяет мосту загружать данные из большого диапазона адресов в кэш сразу, формируя «взрывную» транзакцию. Это значит, что вместо того, чтобы запрашивать каждое значение по отдельности, мост может заранее выбрать и закэшировать данные, которые, вероятно, понадобятся в будущем.

**13. Определение понятия интерфейса.**

Средства, используемые для соединения двух компонентов или систем, называются ***интерфейсом***.

***Интерфейс ЭВМ*** — это совокупность унифицированных аппаратурных, программных и конструктивных средств, необходимых для реализации алгоритмов взаимодействия функциональных устройств.

Интерфейс в общем случае состоит из магистрали и аппаратурных средств (контроллеров, адаптеров), работающих под управлением некоторых программ.

Система сигнальных линий оптимизирована под определенный вид коммуникаций.

Параметры, характеризующие интерфейс:

* Совокупность сигнальных линий;
* Физические, механические и электрические характеристики шины;
* Используемые сигналы арбитража (процедура допуска к управлению шиной только одного из претендентов), состояния, управления и синхронизации;
* Правила взаимодействия подключенных устройств (протокол).

**14. Определение транзакции.**

***Транзакция*** – атомарная операция обмена данными между двумя устройствами PCI

В рамках транзакции определены два объекта:

1. инициатор обмена (Initiator);
2. целевое устройство (Target).

Транзакция состоит из фазы адресациии, одной или нескольких фаз передачи данных.

* Адресация:

Инициатор выставляет адрес целевого устройства и тип операции (чтение или запись).

* Передача данных:

Данные передаются от инициатора к целевому устройству или наоборот.

* Завершение:

Целевое устройство подтверждает завершение транзакции

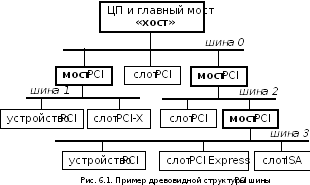
В рамках одной физической шины в конкретный момент может происходить только одна транзакция. Если физических шин несколько, то транзакции на них могут выполняться одновременно, если пути прохождения данных не пересекаются.

На шине PCI отсутствует главный контроллер; любое устройство может стать инициатором. Для выбора инициатора используется арбитраж, который происходит во время выполнения текущей транзакции.

Поддерживаются расщепленные транзакции, когда целевое устройство может занять время для обработки запроса. В этом случае инициатор освобождает шину для других устройств и повторно захватывает ее для завершения транзакции после получения ответа от целевого устройства.

**15. Какие функции выполняют PCI-мосты, и почему они важны в архитектуре компьютера?**

***Мост*** – это чип, который распаян на материнской плате и является частью чипсета.

Мосты играют роль арбитров, обрабатывая запросы от устройств на доступ к шине и отслеживая соблюдение протокола обмена.

Для увеличения числа подключаемых устройств применяют ***мосты РСI*** - устройства РСI с парой интерфейсов, которыми шины объединяются в древовидную структуру.

В корне этой структуры находится ***хост*** — «хозяин шины», в обязанности которого входит конфигурирование всех устройств, включая мосты.

В роли хоста, как правило, выступает центральный процессор с главным мостом.

***Каждый мост*** *PCI соединяет только****две****шины:* ***первичную***, находящуюся ближе к вершине иерархии, с ***вторичной***.

Допускается только чисто ***древовидная конфигурация***. Мосты позволяют объединять шины РСI и РСI-Х с разными характеристиками, а также подключать к РСI/РСI-Х иные шины.

Мост должен выполнять ряд обязательных функций:

-выполнять арбитраж — прием сигналов запроса REQx# от ведущих устройств шины и предоставление им права на управление шиной сигналами GNTx#

-парковать шину — подавать сигнал GNTx# какому-то устройству, когда управление шиной не требуется ни одному из задатчиков;

-определять возможности подключенных устройств и выбирать удовлетворяющий их режим работы шины (частота, разрядность, протокол);

-поддерживать карты ресурсов, находящихся по разные стороны моста;

-отвечать под видом целевого устройства на транзакции, инициированные мастером на одном интерфейсе и адресованные к ресурсу, находящемся со стороны другого интерфейса;

-транслировать эти транзакции на другой интерфейс, выступая в роли ведущего устройства, и передавать их результаты истинному инициатору.