

ВВОД-ВЫВОД 16

Курс лекций

«Системное программное обеспечение»

«System Software»

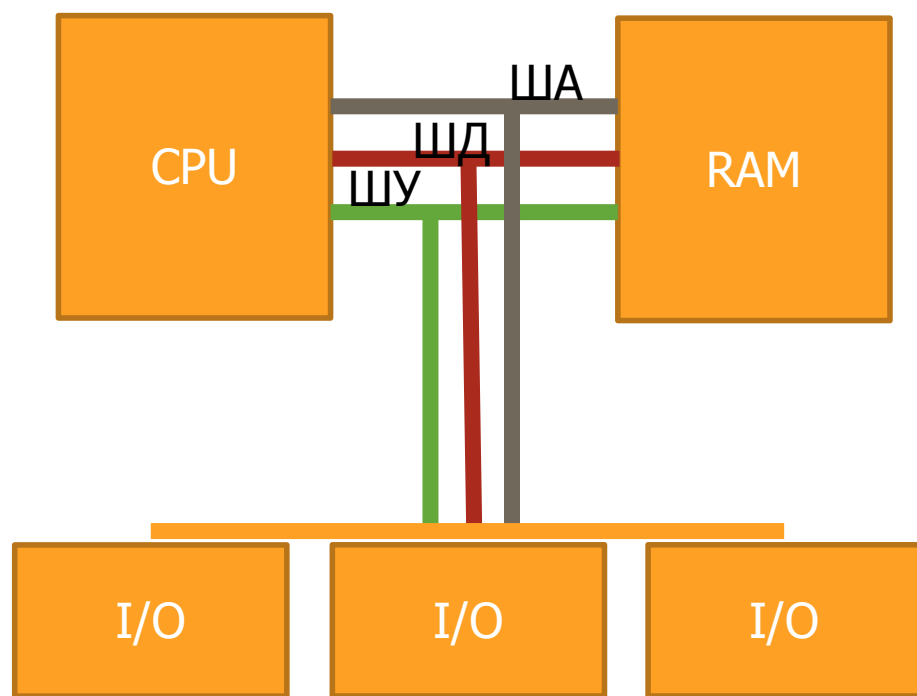
«Операционные системы»

для студентов специальностей АСОИ и ИИ

Павел Кочурко
доцент кафедры ИИТ, к.т.н.



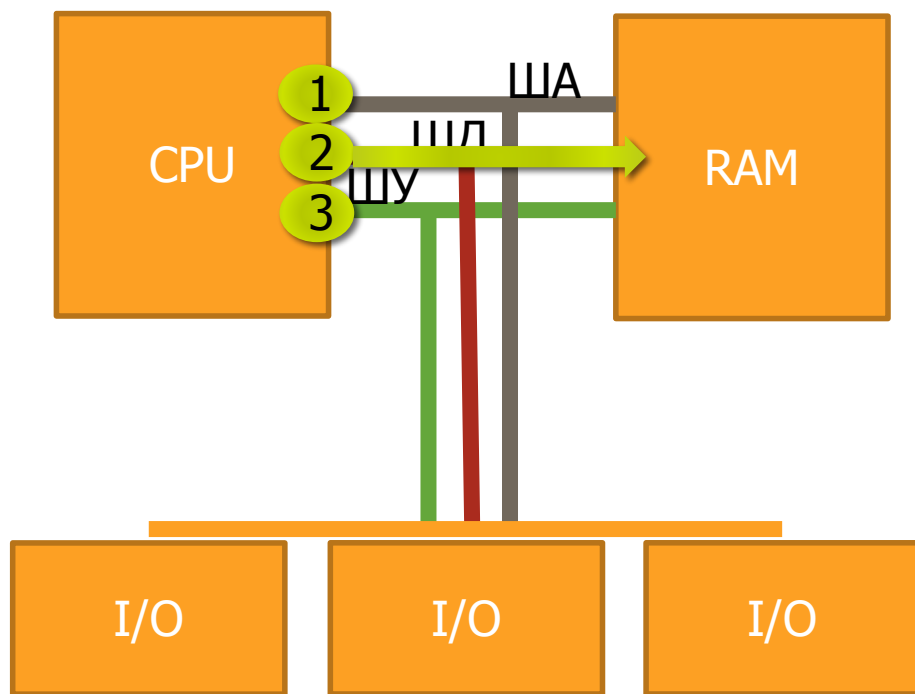
Физическая организация ввода-вывода



Локальная магистраль компьютера – **линии**, объединённые в **шины**

- **шина данных** - для передачи информации между процессором и памятью, процессором и устройствами ввода-вывода, памятью и внешними устройствами;
- **адресная шина** - для задания адреса ячейки памяти или указания устройства ввода-вывода, участвующих в обмене информацией;
- **шина управления** - определяет поведение локальной магистрали

Запись в память



1. На адресной шине процессор должен выставить сигналы, соответствующие адресу ячейки памяти, в которую будет осуществляться передача информации.
2. На шину данных процессор должен выставить сигналы, соответствующие информации, которая должна быть записана в память.
3. После выполнения действий 1 и 2 на шину управления выставляются сигналы, соответствующие операции записи и работе с памятью, что приведет к **занесению необходимой информации по нужному адресу.**

Порты ввода-вывода

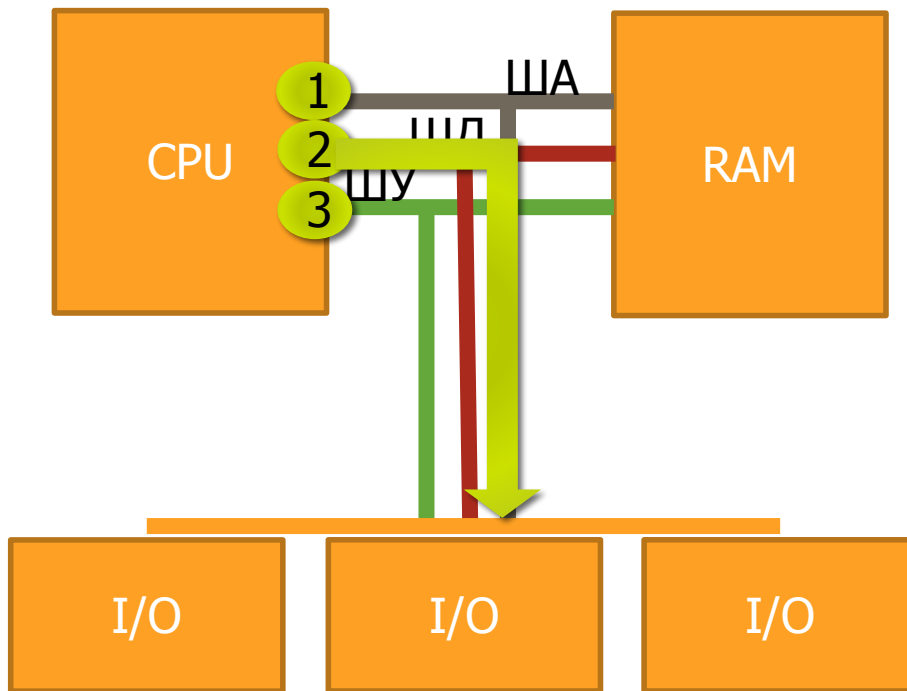
Внешние устройства разнесены пространственно и могут подключаться к локальной магистрали в одной точке или множестве точек, получивших название **портов ввода-вывода**.

Порты ввода-вывода взаимно однозначно отображаются в **адресное пространство ввода-вывода**. При этом каждый порт ввода-вывода получает свой номер или адрес в этом пространстве.

Если адресное пространство памяти (размер которого определяется шириной адресной шины) задействовано не полностью (остались адреса, которым не соответствуют физические ячейки памяти) и протоколы работы с внешним устройством совместимы с протоколами работы с памятью, часть портов ввода-вывода может быть отображена непосредственно в адресное пространство памяти



Вывод на устройство (упрощённо)



1. На адресной шине процессор должен выставить сигналы, соответствующие адресу порта, в который будет осуществляться передача информации, в адресном пространстве ввода-вывода.
2. На шину данных процессор должен выставить сигналы, соответствующие информации, которая должна быть передана в порт.
3. После выполнения действий 1 и 2 на шину управления выставляются сигналы, соответствующие операции записи и **работе с устройствами ввода-вывода (переключение адресных пространств!)**, что приведет к передаче необходимой информации в нужный порт.

Передача данных в порт ввода-вывода только начинает длительный процесс вывода

Вывод

- Устройства ввода-вывода подключаются к системе через порты.
- Могут существовать два адресных пространства: пространство памяти и пространство ввода-вывода.
- Порты, как правило, отображаются в адресное пространство ввода-вывода и иногда – непосредственно в адресное пространство памяти.
- Использование того или иного адресного пространства определяется типом команды, выполняемой процессором, или типом ее операндов.
- Физическим управлением устройством ввода-вывода, передачей информации через порт и выставлением некоторых сигналов на магистрали занимается контроллер устройства.

Контроллеры устройств ввода-вывода

Контроллер – устройство, непосредственно подключенное к локальной магистрали и управляющее устройством ввода-вывода. Четыре внутренних регистра, для доступа к которым могут использоваться один или несколько портов:

- **Регистр состояния**
содержит биты, значение которых определяется состоянием устройства ввода-вывода и которые доступны только для чтения вычислительной системой (бит занятости, ошибки, готовности...)
- **Регистр управления**
получает данные, которые записываются вычислительной системой для инициализации устройства ввода-вывода или выполнения очередной команды, а также изменения режима работы устройства
- **Регистр выходных данных**
для помещения в него данных для чтения вычислительной системой
- **Регистр входных данных**
для помещения в него информации, которая должна быть выведена на устройство

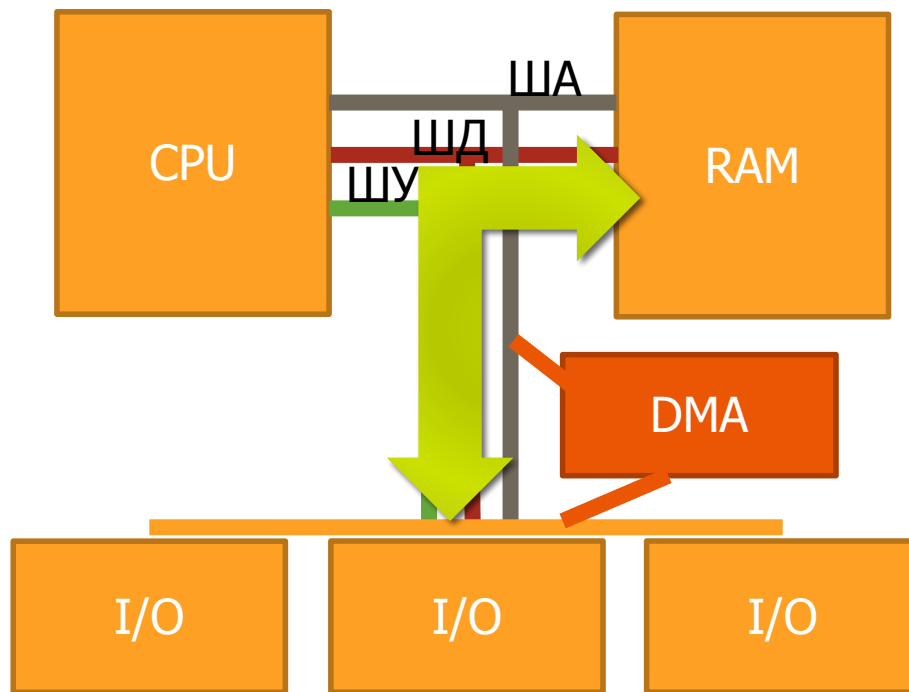


Вывод на устройство (детально)

1. Процессор в цикле читает информацию из порта регистра состояний и проверяет значение **бита занятости**. Если **бит занятости** установлен, то это означает, что устройство еще не завершило предыдущую операцию, и процессор уходит на новую итерацию цикла. Если **бит занятости** сброшен, то устройство готово к выполнению новой операции, и процессор переходит на следующий шаг.
2. Процессор записывает код команды вывода в порт регистра управления.
3. Процессор записывает данные в порт регистра входных данных.
4. Процессор устанавливает **бит готовности команды**. В следующих шагах процессор не задействован.
5. Когда контроллер замечает, что **бит готовности команды** установлен, он устанавливает **бит занятости**.
6. Контроллер анализирует код команды в регистре управления и обнаруживает, что это команда вывода. Он берет данные из регистра входных данных и инициирует выполнение команды.
7. После завершения операции контроллер обнуляет **бит готовности команды**.
8. При успешном завершении операции контроллер обнуляет **бит ошибки** в регистре состояния, при неудачном завершении команды – устанавливает его.
9. Контроллер сбрасывает **бит занятости**.



Прямой доступ к памяти



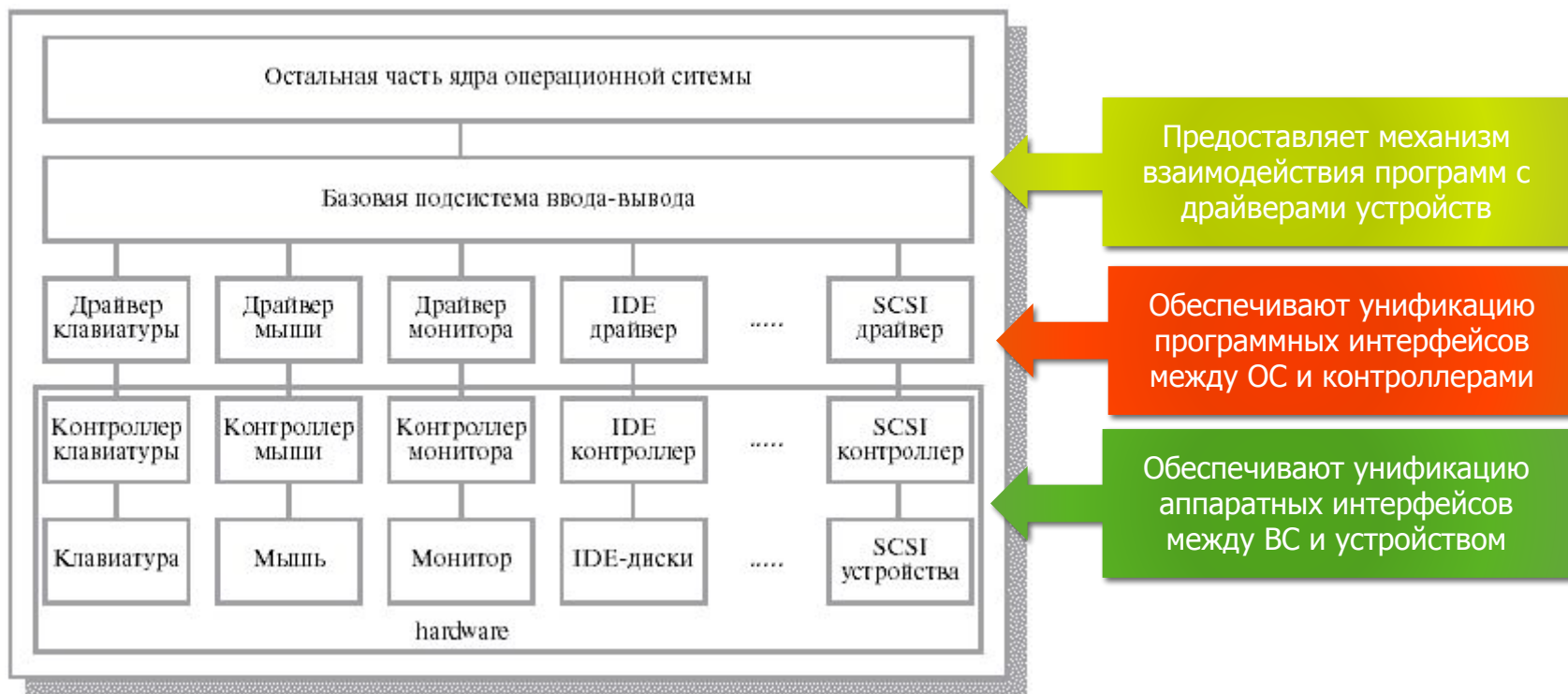
Direct Memory Access (DMA)

Контроллер DMA имеет несколько спаренных линий – каналов DMA, которые могут подключаться к различным устройствам.

1. Получив по одному из каналов DMA сигнал запроса на передачу данных от внешнего устройства, контроллер по шине управления сообщает процессору о желании взять на себя управление локальной магистралью.
2. Процессор передает управление ею контроллеру DMA.
3. Контроллер DMA выставляет на адресную шину адрес памяти для передачи очередной порции информации и по второй линии канала прямого доступа к памяти сообщает устройству о готовности магистрали к передаче данных.
4. После этого, используя шину данных и шину управления, контроллер DMA, устройство ввода-вывода и память осуществляют процесс обмена информацией.
5. Затем контроллер прямого доступа к памяти извещает процессор о своем отказе от управления магистралью, и тот берет руководящие функции на себя.

+ : CPU не занимается управлением передачей больших объемов данных между памятью и устройствами ввода-вывода вместо полезной вычислительной работы

Структура системы ввода-вывода



Функции БПВВ (BIOS)

Базовая подсистема ввода-вывода служит посредником между процессами вычислительной системы и набором драйверов.

Системные вызовы для выполнения операций ввода-вывода трансформируются ею в вызовы функций необходимого драйвера устройства

- поддержка блокирующихся, неблокирующихся и асинхронных системных вызовов
- буферизация и кэширование входных и выходных данных
- осуществление спулинга и монопольного захвата внешних устройств
- обработка ошибок и прерываний, возникающих при операциях ввода-вывода
- планирование последовательности запросов на выполнение этих операций

Классификации устройств ввода-вывода

- По скорости обмена информацией: от нескольких байтов в секунду (клавиатура) до нескольких гигабайтов в секунду (сетевые карты).
- Могут использоваться несколькими процессами параллельно (являются разделяемыми), в то время как другие требуют монопольного захвата процессом.
- Устройства могут запоминать выведенную информацию для ее последующего ввода или не обладать этой функцией. Устройства, запоминающие информацию, в свою очередь, могут дифференцироваться по формам доступа к сохраненной информации: обеспечивать к ней последовательный доступ в жестко заданном порядке или уметь находить и передавать только необходимую порцию данных.
- Часть устройств умеет передавать данные только по одному байту последовательно (символьные устройства), а часть устройств умеет передавать блок байтов как единое целое (блочные устройства).
- Предназначенные только для ввода информации, предназначенные только для вывода информации, и устройства, которые могут выполнять и ввод, и вывод.

ВОПРОСЫ?

<http://iit.bstu.by/ss>

