

# **Основные понятия микропроцессорных систем**

Материалы по дисциплине «Микропроцессорные системы»  
Специальность «Компьютерные системы и комплексы»  
Составитель: Торгашин Р.Г

ГБПОУ ВО "Борисоглебский техникум промышленных и информационных технологий"

2016 год

## Оглавление

Архитектура и Семейство.....	3
Семейства вычислительных систем.....	3
Типы микропроцессорных систем.....	5
Основные архитектуры вычислительных машин.....	8
Архитектура фон Неймана.....	8
Гарвардская архитектура.....	10
Структура связей в микропроцессорной системе.....	11
Системная шина (магистраль).....	14
Структура системной шины.....	14
Работа микропроцессорной системы с шиной.....	17
Прохождение сигналов по магистрали.....	18
Вопросы для контроля.....	21
Источники:.....	22

## Архитектура и Семейство

**Архитектура** – это наиболее общие принципы построения ЭВМ, реализующие программное управление работой и взаимодействием основных ее функциональных узлов

Архитектура ЭВМ используется для описания принципа действия, конфигурации и взаимного соединения основных логических узлов ЭВМ (вследствие чего термин “архитектура” оказывается ближе к обыденному значению этого слова).

С точки зрения архитектуры представляют интерес лишь те связи и принципы, которые являются наиболее общими, присущими многим конкретным реализациям вычислительных машин.

Общие принципы построения ЭВМ, которые относятся к архитектуре:

- структура памяти ЭВМ;
- способы доступа к памяти и внешним устройствам;
- система команд;
- форматы данных;
- организация интерфейса.
- возможность изменения конфигурации компьютера;

Расширение сферы применения вычислительной техники, особенно их применение для решения научно-технических задач, задач автоматизации производства, САПР и т.д. привело к необходимости создания вычислительных систем, отличающихся составом технических средств (ТС) и техническими характеристиками: производительностью, емкостью оперативной памяти (ОП), внешних запоминающих устройств (ВЗУ).

В то же время, разрабатывать новую архитектуру для каждого класса решаемых задач нерентабельно. Отсюда следует, что нужна ЭВМ с единой архитектурой, но с переменным составом оборудования. Такой подход означает выполнение отдельных функциональных устройств в виде модулей, которые могут объединяться в необходимом количестве в одной ЭВМ. Подобрать необходимые модули можно создать ЭВМ наиболее подходящую для решения возлагаемых на нее задач.

Очевидно, что для того, чтобы модули составляющие ЭВМ могли взаимодействовать, они должны соответствовать единым требованиям. **О совместимых ЭВМ говорят что они принадлежат к одному семейству.** А компоненты, которые можно использовать в ЭВМ определенного семейства — совместимы с этим семейством. При этом важным является сокращение числа типов (номенклатуры) выпускаемых семейств ЭВМ.

### Семейства вычислительных систем.

В пределах одного семейства основные принципы устройства и функционирования машин одинаковы, хотя отдельные модели могут существенно различаться по производительности, стоимости и другим параметрам.

Принадлежность к одному семейству для систем с различным составом оборудования, производительностью и ценой определяется по:

- информационной совместимости;
- программной совместимости снизу-вверх и сверху-вниз;
- аппаратной совместимости;
- конструктивной совместимости;
- эксплуатационной совместимости.

**Информационная совместимость** ЭВМ предполагает использование единых способов

кодирования информации, форматов и типов данных, одинаковые или кратные длины машинных слов в различных моделях. Эта совместимость является базовой, без ее обеспечения подключение устройств возможно только с использованием контроллеров сопряжения

**Программная совместимость** означает, что программы, написанные для одной модели, должны выполняться для других моделей семейства. Это предполагает наличие единой системы и форматов команд, режимов адресации, что позволяет использовать общие ОС и прикладное ПО для моделей одного семейства снизу доверху.

Например системные требования новейшей на 2013 год операционной системы Windows8<sup>1</sup>:

Processor: 1 gigahertz (GHz) or faster

- RAM: 1 gigabyte (GB) (32-bit) or 2 GB (64-bit)
- Free hard disk space: 16 GB (32-bit) or 20 GB (64-bit)
- Graphics card: Microsoft DirectX 9 graphics device with WDDM driver

Программной совместимостью объясняется наличие большого числа форматов команд и системы команд для старших моделей МКПР, что затрудняет их изучение.

**Аппаратная совместимость** заключается в возможности подключения к любой модели ЭВМ, состоящей из центрального процессора и оперативной памяти любых контроллеров периферийных устройств (ПУ), общих для всех моделей ряда. Это достигается за счет использования унифицированных интерфейсов ввода-вывода и единых протоколов обмена между устройствами и центральным процессором. Следует заметить, что практически каждая новая модель микропроцессора имеет свой состав аппаратных средств, которые программно и аппаратно совместимы с предыдущими версиями.

Фактически, аппаратная совместимость обеспечивает единство протоколов и параметров разъемов.

**Конструктивная совместимость** подразумевает использование унифицированных панелей, блоков и ТЭЗов (плат) с единой системой назначения контактов разъемов и типов разъемов, конструктивного исполнения системного блока и разбивки ТС на конструктивные модули. Подобная совместимость гарантирует, что устройство можно механически подключить к системе. Все разъемы и крепления подойдут

**Эксплуатационная совместимость** предполагает общие методы технической и математической эксплуатации и обслуживания, т.е. преемственность языков программирования, единых ОС, программ технического обслуживания и диагностики, единые методы профилактики ТС и т.д., что не требует переквалификации и дополнительного обучения обслуживающего персонала.

Все рассмотренные особенности и требования к семействам ЭВМ позволяют создавать системы переменной конфигурации с возможностью постепенного, по мере необходимости, наращивания вычислительной мощности путем замены процессора более производительным, расширения емкости оперативной памяти, подключения новых и замены устаревших периферийных устройств. При этом за счет программной совместимости моделей одного семейства ЭВМ все ранее существующее программное обеспечение сохраняется.

Улучшение технико-экономических характеристик моделей семейства ведется в следующих на-правлениях:

- совершенствования элементной базы;
- повышения производительности за счет применения новых технологий и структурных решений;
- увеличения объема оперативной памяти и внешних запоминающих устройств, совершенствования организации хранения данных, реализации виртуальной памяти, использования как внутренней на кристалле, так и внешней кэш-памяти;
- дальнейшего развития системы программного обеспечения как в области прикладных

---

1 <http://windows.microsoft.com/en-us/windows-8/preview-faq>

- программ, так и ОС;
- создания мультипрограммных и многомашинных вычислительных систем, работающих в реальном времени, локальных и глобальных вычислительных систем;
- развития системы ввода-вывода и расширения номенклатуры периферийных устройств;
- повышения надежности, развития эффективных систем контроля и диагностики;
- за счет совершенствования технологии.

Технические средства семейств ЭВМ постоянно совершенствуются и развиваются в следующих направлениях:

- модификации - созданию нескольких моделей одного и того же устройства, отличающихся друг от друга значением какого-либо параметра или набором функции. Как правило, модификации появляются вследствие доработки предыдущих версий устройства для устранения каких-либо недостатков или дополнения функциональных возможностей устройства без изменения его технических характеристик.
- модернизации, т.е. замены устаревших образцов устройств новыми с улучшенными техническими, функциональными, конструктивными и эксплуатационными характеристиками;
- создания принципиально новых устройств. Например, лазерный принтер является принципиально новой разработкой среди принтеров, как по принципу функционирования, так и по техническим характеристикам.

## Типы микропроцессорных систем

Диапазон применения микропроцессорной техники сейчас очень широк, требования к микропроцессорным системам предъявляются самые разные. Поэтому сформировалось несколько типов микропроцессорных систем, различающихся мощностью, универсальностью, быстродействием и структурными отличиями. Основные типы следующие:

- микроконтроллеры — наиболее простой тип микропроцессорных систем, в которых все или большинство узлов системы выполнены в виде одной микросхемы;
- контроллеры — управляющие микропроцессорные системы, выполненные в виде отдельных модулей;
- микрокомпьютеры — более мощные микропроцессорные системы с развитыми средствами сопряжения с внешними устройствами.
- компьютеры (в том числе персональные) — самые мощные и наиболее универсальные микропроцессорные системы.

Четкую границу между этими типами иногда провести довольно сложно. Быстродействие всех типов микропроцессоров постоянно растет, и нередки ситуации, когда новый микроконтроллер оказывается быстрее, например, устаревшего персонального компьютера.

**Микроконтроллеры** представляют собой универсальные устройства, которые практически всегда используются не сами по себе, а в составе более сложных устройств, в том числе и контроллеров. Системная шина микроконтроллера скрыта от пользователя внутри микросхемы. Возможности подключения внешних устройств к микроконтроллеру ограничены. Устройства на микроконтроллерах обычно предназначены для решения одной задачи



Рис. 25 Микроконтроллер Atmel AVR ATmega169V в корпусе MLF

**Контроллеры**, как правило, создаются для решения какой-то отдельной задачи или группы близких задач. Они обычно не имеют возможностей подключения дополнительных узлов и устройств, например, большой памяти, средств ввода/вывода. Их системная шина чаще всего недоступна пользователю. Структура контроллера проста и оптимизирована под максимальное быстродействие. В большинстве случаев выполняемые программы хранятся в постоянной памяти и не меняются. Конструктивно контроллеры выпускаются в одноплатном варианте.



Рис. 26 Контроллер Controller ASR-USB 3.0 CARD Asrock 2x USB 3.0 ports PCX1

**Микрокомпьютеры** отличаются от контроллеров более открытой структурой, они допускают подключение к системной шине нескольких дополнительных устройств. Производятся микрокомпьютеры в корпусе, корпусе с разъемами системной магистрали, доступными пользователю. Микрокомпьютеры могут иметь средства хранения информации на магнитных носителях (например, магнитные диски) и довольно развитые средства связи с пользователем (видеомонитор, клавиатура). Микрокомпьютеры рассчитаны на широкий круг задач, но в отличие от контроллеров, к каждой новой задаче его надо приспособлять заново. Выполняемые микрокомпьютером программы можно легко менять.



Рис. 27 Микрокомпьютер Raspberry Pi model B ARM11/512 Мб (2 190 р.)

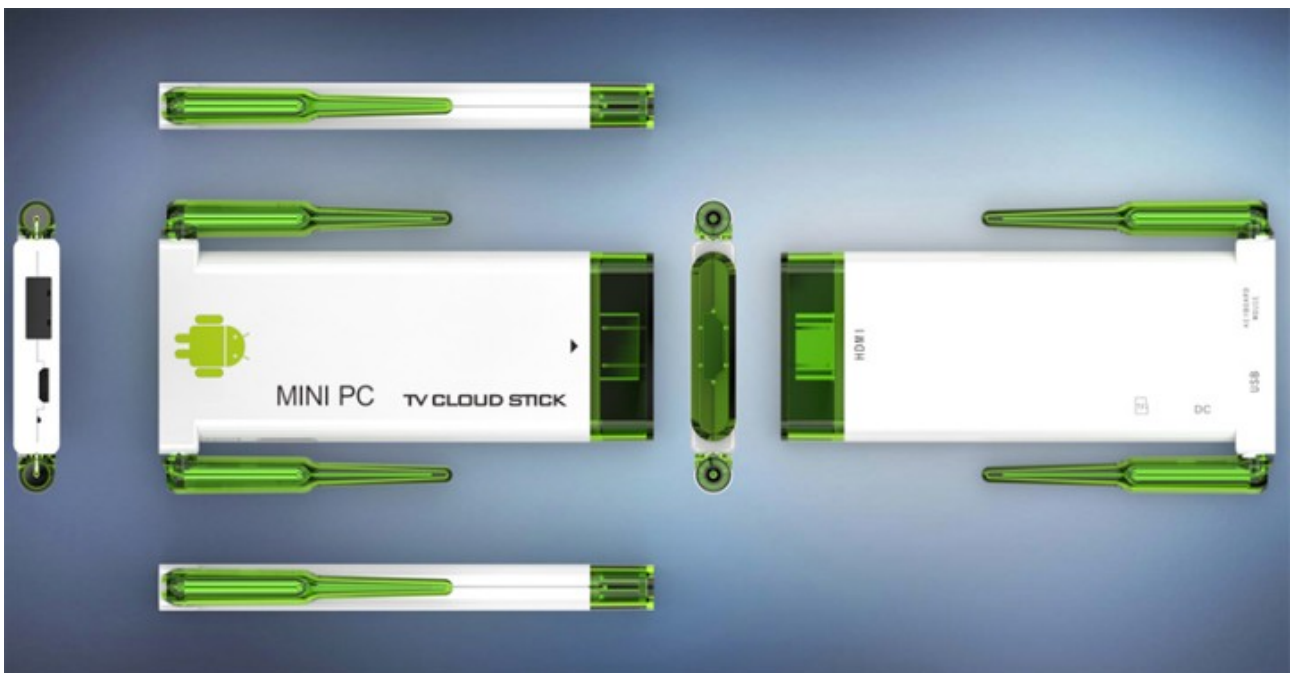


Рис. 28 Микрокомпьютер DM918 Android TV Stick (70 \$)

**Компьютеры** и самые распространенные из них — персональные компьютеры — это самые универсальные из микропроцессорных систем. Они обязательно предусматривают возможность модернизации, а также широкие возможности подключения новых устройств. Их системная шина, конечно, доступна пользователю. Кроме того, внешние устройства могут подключаться к компьютеру через несколько встроенных портов связи (количество портов доходит иногда до 10). Компьютер всегда имеет сильно развитые средства связи с пользователем, средства длительного хранения информации большого объема, средства связи с другими компьютерами по информационным сетям. Области применения компьютеров могут быть самыми разными: математические расчеты, обслуживание доступа к базам данных, управление работой сложных электронных систем, компьютерные игры, подготовка документов и т. д.



**Рис. 29** Компьютер Lenovo IdeaCentre Q190 (57-312200) Celeron 887 1.5 DDR3 2GB 320GB HDMI/VGA звук 7.1 4USB2.0/2USB3.0 LAN1Gb WiFi MMC/MS/MS Pro/SD/SDHC/SDXC черный-серебристый DOS

Любую задачу в принципе можно выполнить с помощью каждого из перечисленных типов микропроцессорных систем. Но при выборе типа надо по возможности избегать избыточности и предусматривать необходимую для данной задачи гибкость системы. В настоящее время при разработке новых микропроцессорных систем чаще всего выбирают путь использования микроконтроллеров (примерно в 80% случаев). При этом микроконтроллеры применяются или самостоятельно, с минимальной дополнительной аппаратурой, или в составе более сложных контроллеров с развитыми средствами ввода/вывода.

Классические микропроцессорные системы на базе микросхем процессоров и микропроцессорных комплектов выпускаются сейчас довольно редко, в первую очередь, из-за сложности процесса разработки и отладки этих систем. Данный тип микропроцессорных систем выбирают в основном тогда, когда микроконтроллеры не могут обеспечить требуемых характеристик.

Наконец, заметное место занимают сейчас микропроцессорные системы на основе персонального компьютера. Разработчику в этом случае нужно только оснастить персональный компьютер дополнительными устройствами сопряжения, а ядро микропроцессорной системы уже готово. Персональный компьютер имеет развитые средства программирования, что существенно упрощает задачу разработчика. К тому же он может обеспечить самые сложные алгоритмы обработки информации. Основные недостатки персонального компьютера — большие размеры корпуса и аппаратурная избыточность для простых задач. Недостатком является и неприспособленность большинства персональных компьютеров к работе в сложных условиях (запыленность, высокая влажность, вибрации, высокие температуры и т.д.). Однако выпускаются и специальные персональные компьютеры, приспособленные к различным условиям эксплуатации.

## **Основные архитектуры вычислительных машин**

### ***Архитектура фон Неймана***

Основы учения об архитектуре вычислительных машин заложил выдающийся американский математик Джон фон Нейман. Он подключился к созданию первой в мире ламповой ЭВМ ENIAC в 1944 г., когда ее конструкция была уже выбрана. В процессе работы



во время многочисленных дискуссий со своими коллегами Г. Голдстайном и А. Берксом фон Нейман высказал идею принципиально новой ЭВМ. В 1946 г. ученые изложили свои принципы построения вычислительных машин в ставшей классической статье “Предварительное рассмотрение логической конструкции электронно-вычислительного устройства”. С тех пор прошло полвека, но выдвинутые в ней положения сохраняют актуальность и сегодня.

В статье фон Неймана убедительно обосновывается использование двоичной системы для представления чисел

Фон Нейман не только выдвинул основополагающие принципы логического устройства ЭВМ, но и предложил ее структуру, которая воспроизводилась в течение первых двух поколений ЭВМ. Основными блоками по Нейману являются устройство управления (УУ) и арифметико-логическое устройство (АЛУ) (обычно объединяемые в центральный процессор), память, внешняя память, устройства ввода и вывода. Схема устройства такой ЭВМ представлена на рис. 1. Следует отметить, что внешняя память отличается от устройств ввода и вывода тем, что данные в нее заносятся в виде, удобном компьютеру, но недоступном для непосредственного восприятия человеком. Так, накопитель на магнитных дисках относится к внешней памяти, а клавиатура – устройство ввода, дисплей и печать – устройства вывода.

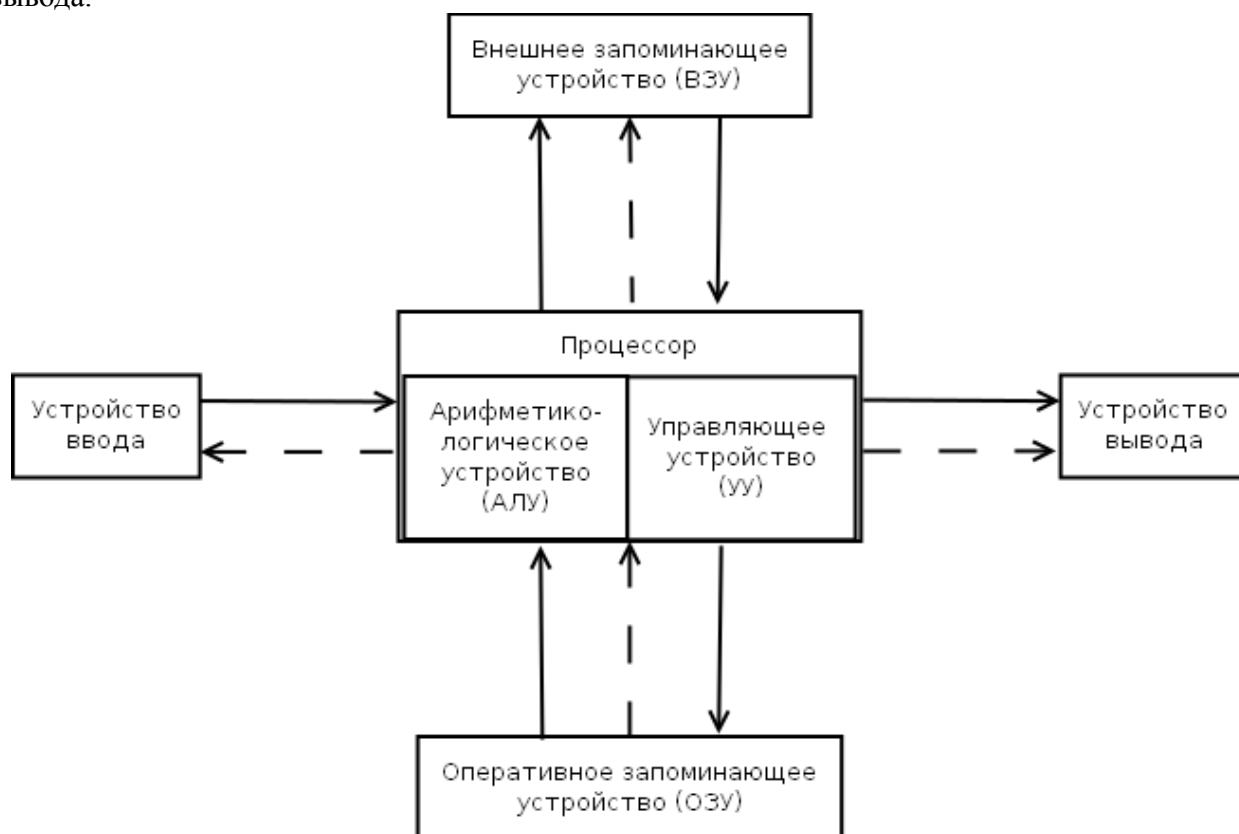


Рисунок 1 Архитектура Фон Неймана

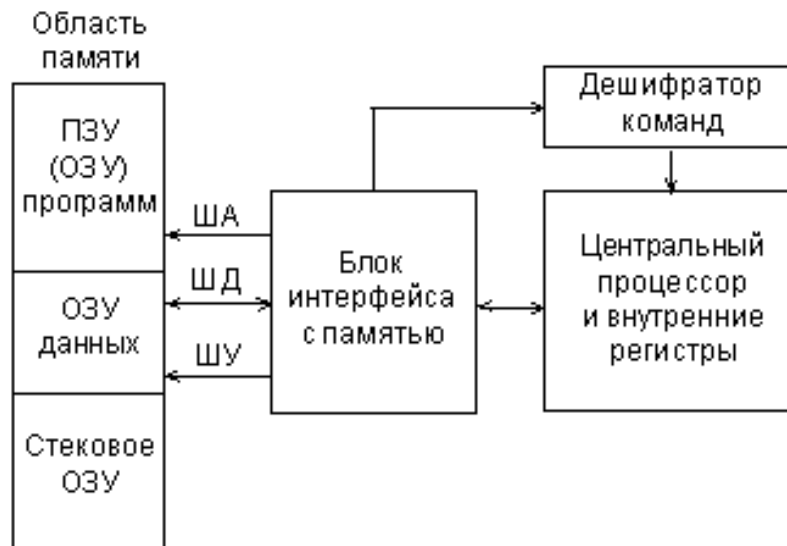


Рисунок 2 Структура микропроцессорной системы с Фон-Немановской архитектурой

Устройство управления и арифметико-логическое устройство в современных компьютерах объединены в один блок – процессор, являющийся преобразователем информации, поступающей из памяти и внешних устройств (сюда относятся выборка команд из памяти, кодирование и декодирование, выполнение различных, в том числе и арифметических, операций, согласование работы узлов компьютера).

Память (ЗУ) хранит информацию (данные) и программы. Запоминающее устройство у современных компьютеров “многоярусно” и включает оперативное запоминающее устройство (ОЗУ), хранящее ту информацию, с которой компьютер работает непосредственно в данное время (исполняемая программа, часть необходимых для нее данных, некоторые управляющие программы), и внешние запоминающие устройства (ВЗУ) гораздо большей емкости, чем ОЗУ, но с существенно более медленным доступом (и значительно меньшей стоимостью в расчете на 1 байт хранимой информации). На ОЗУ и ВЗУ классификация устройств памяти не заканчивается – определенные функции выполняют и СОЗУ (сверхоперативное запоминающее устройство, КЭШ), и ПЗУ (постоянное запоминающее устройство), и другие подвиды компьютерной памяти.

В построенной по описанной схеме ЭВМ происходит последовательное считывание команд из памяти и их выполнение. Номер (адрес) очередной ячейки памяти, из которой будет извлечена следующая команда программы, указывается специальным устройством – счетчиком команд в УУ. Его наличие также является одним из характерных признаков рассматриваемой архитектуры.

Разработанные фон Нейманом основы архитектуры вычислительных устройств оказались настолько фундаментальными, что получили в литературе название “фон-неймановской архитектуры”. Подавляющее большинство вычислительных машин на сегодняшний день – фон-неймановские машины. Исключение составляют лишь отдельные разновидности систем для параллельных вычислений, в которых отсутствует счетчик команд, не реализована классическая концепция переменной и имеются другие существенные принципиальные отличия от классической модели (примерами могут служить потоковая и редуцированная вычислительные машины).

## Гарвардская архитектура

Основной особенностью гарвардской архитектуры является использование отдельных

адресных пространств для хранения команд и данных, как показано на рисунке 3

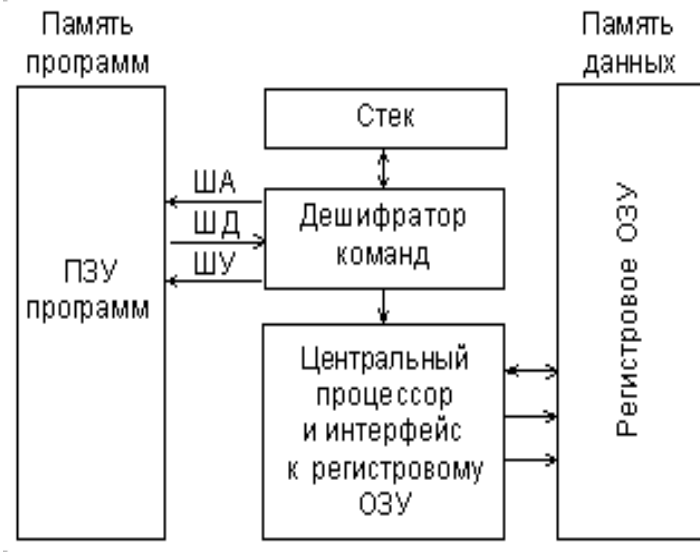


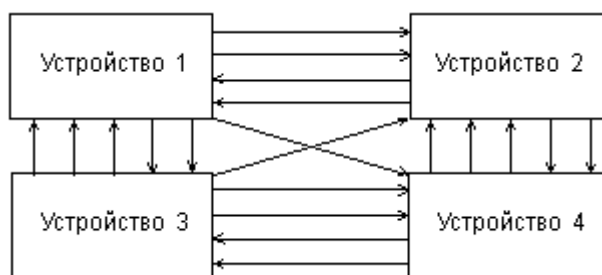
Рисунок 3 Структура микропроцессорной системы с гарвардской архитектурой

Гарвардская архитектура почти не использовалась до конца 70-х годов, пока производители МК не поняли, что она дает определенные преимущества разработчикам автономных систем управления.

*Судя по опыту использования МПС для управления различными объектами, для реализации большинства алгоритмов управления такие преимущества фон-неймановской архитектуры как гибкость и универсальность не имеют большого значения. Анализ реальных программ управления показал, что необходимый объем памяти данных МК, используемый для хранения промежуточных результатов, как правило, на порядок меньше требуемого объема памяти программ. В этих условиях использование единого адресного пространства приводило к увеличению формата команд за счет увеличения числа разрядов для адресации операндов. Применение отдельной небольшой по объему памяти данных способствовало сокращению длины команд и ускорению поиска информации в памяти данных.*

Гарвардская архитектура обеспечивает потенциально более высокую скорость выполнения программы по сравнению с фон-неймановской за счет возможности реализации параллельных операций. Выборка следующей команды может происходить одновременно с выполнением предыдущей, и нет необходимости останавливать процессор на время выборки команды. Этот метод реализации операций позволяет обеспечивать выполнение различных команд за одинаковое число тактов, что дает возможность более просто определить время выполнения циклов и критичных участков программы.

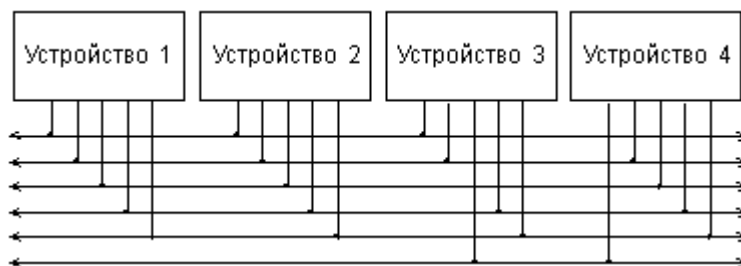
## Структура связей в микропроцессорной системе



**Рисунок 4. Классическая структура связей.**

При классической структуре связей (рис. 4) все сигналы и коды между устройствами передаются по отдельным линиям связи. Каждое устройство, входящее в систему, передает свои сигналы и коды независимо от других устройств. При этом в системе получается очень много линий связи и разных протоколов обмена информацией. Такую систему сложно проектировать и реализовать. Причем сложность растет экспоненциально, по мере увеличения числа устройств. Это затрудняет масштабирование.

При шинной структуре связей (рис. 4) все сигналы между устройствами передаются по одним и тем же линиям связи, но в разное время (это называется мультиплексированной передачей). Причем передача по всем линиям связи может осуществляться в обоих направлениях (так называемая двунаправленная передача). В результате количество линий связи существенно сокращается, а правила обмена (протоколы) упрощаются. Группа линий связи, по которым передаются сигналы или коды как раз и называется шиной (англ. Bus).



**Рисунок 5. Шинная структура связей.**

Понятно, что при шинной структуре связей легко осуществляется пересылка всех информационных потоков в нужном направлении, например, их можно пропустить через один процессор, что очень важно для микропроцессорной системы. Однако при шинной структуре связей вся информация передается по линиям связи последовательно во времени, по очереди, что снижает быстродействие системы по сравнению с классической структурой связей.

Большое достоинство шинной структуры связей состоит в том, что все устройства, подключенные к шине, должны принимать и передавать информацию по одним и тем же правилам (протоколам обмена информацией по шине). Соответственно, все узлы, отвечающие за обмен с шиной в этих устройствах, должны быть единообразны, унифицированы.

Существенный недостаток шинной структуры связан с тем, что все устройства подключаются к каждой линии связи параллельно. Поэтому любая неисправность любого устройства может вывести из строя всю систему, если она портит линию связи. Например, некоторые неисправности оперативной памяти и жестких дисков вызывают передачу по шине большого числа «паразитных» данных. Что, в свою очередь становится причиной торможения или зависания системы.

По этой же причине отладка системы с шинной структурой связей довольно сложна и обычно требует специального оборудования. В системах с шинной структурой связей применяют все три существующие разновидности

выходных каскадов цифровых микросхем:

- стандартный выход или выход с двумя состояниями (обозначается 2С, 2S, реже ТТЛ, TTL);
- выход с открытым коллектором (обозначается ОК, ОС);
- выход с тремя состояниями или (что то же самое) с возможностью отключения (обозначается 3С, 3S).

Упрощенно эти три типа выходных каскадов могут быть представлены в виде схем на рис. 6

**У выхода 2С** два ключа замыкаются по очереди, что соответствует уровням логической единицы (верхний ключ замкнут) и логического нуля (нижний ключ замкнут).

**У выхода ОК** замкнутый ключ формирует уровень логического нуля, разомкнутый — логической единицы.

**У выхода 3С** ключи могут замыкаться по очереди (как в случае 2С), а могут размыкаться одновременно, образуя третье, высокоимпедансное<sup>2</sup> состояние. Переход в третье состояние (Z-состояние) управляется сигналом на специальном входе EZ.

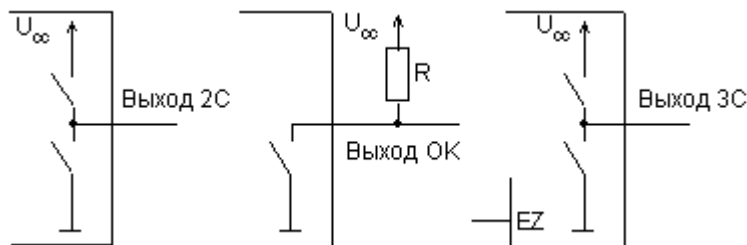


Рисунок 6 Три типа выходов цифровых микросхем.

*В цифровой электронике существуют понятия «логическая единица» (контакт присоединён к источнику питания и может выдавать в нагрузку большой ток, порядка сотен миллиампер) и «логический ноль» (контакт присоединён к «земле», и также выдерживает высокие токи). Но такие выходы нельзя объединять: если на одном будет 1, а на другом 0, возникает короткое замыкание, чреватое выгоранием выходных транзисторов.*

*Поэтому, чтобы можно было организовывать соединение типа «шина», было введено третье «высокоимпедансное состояние», когда дополнительный ключ просто отключает выход и он «повисает в воздухе» — соединяется с остальной схемой через высокое сопротивление (импеданс) закрытого транзистора. Такой выход не влияет на подключённый к нему провод, следовательно к одному проводу можно подключать несколько выходов, нужно только следить, чтобы в каждый момент времени только один был активным, а остальные в высокоимпедансном состоянии. Получается соединение типа «шина».*

Выходные каскады типов 3С и ОК позволяют объединять несколько выходов микросхем для получения мультиплексированных (рис. 7) или двунаправленных (рис. 8) линий.

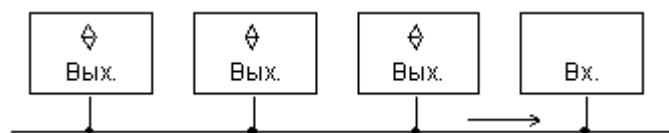


Рисунок 7 Мультиплексированная линия.

<sup>2</sup>Высокоимпедансное состояние или Z-состояние — такое состояние контакта логической схемы, при котором сопротивление между этим контактом и остальной схемой очень велико. Физически реализуется закрытым транзистором, работающим в ключевом режиме.

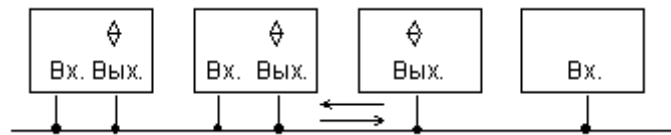


Рисунок 8 Двухнаправленная линия.

При этом в случае выходов 3С необходимо обеспечить, чтобы на линии всегда работал только один активный выход, а все остальные выходы находились бы в это время в третьем состоянии, иначе возможны конфликты. Объединенные выходы ОК могут работать все одновременно, без всяких конфликтов.

## Системная шина (магистраль)

### Структура системной шины

Типичная структура микропроцессорной системы приведена на рис. 9 Она включает в себя три основных типа устройств:

- процессор;
- память, включающую оперативную память ( ОЗУ, RAM — Random Access Memory ) и постоянную память ( ПЗУ, ROM — Read Only Memory ), которая служит для хранения данных и программ;
- устройства ввода/вывода ( УВВ, I/O — Input/Output Devices ), служащие для связи микропроцессорной системы с внешними устройствами, для приема (ввода, чтения, Read) входных сигналов и выдачи (вывода, записи, Write) выходных сигналов.

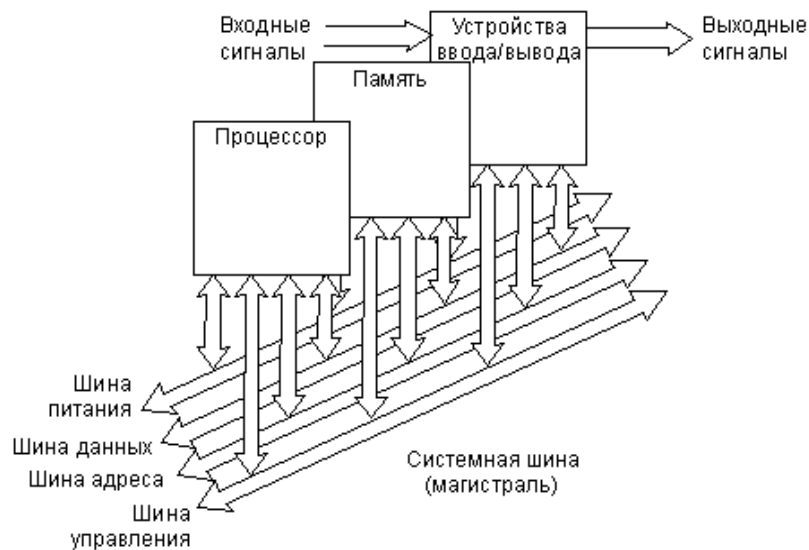


Рисунок 9 Структура микропроцессорной системы.

Все устройства микропроцессорной системы объединяются общей системной шиной (она же называется еще системной магистралью или каналом ). Системная магистраль включает в себя четыре основные шины нижнего уровня:

- шина адреса (Address Bus);
- шина данных (Data Bus);
- шина управления (Control Bus);
- шина питания (Power Bus).

**Шина адреса** служит для определения адреса (номера) устройства, с которым

процессор обменивается информацией в данный момент. Каждому устройству (кроме процессора), каждой ячейке памяти в микропроцессорной системе присваивается собственный адрес. Когда код какого-то адреса выставляется процессором на шине адреса, устройство, которому этот адрес приписан, понимает, что ему предстоит обмен информацией. Шина адреса может быть однонаправленной или двунаправленной.

Шина адреса — вторая по важности шина, которая определяет максимально возможную сложность микропроцессорной системы, то есть допустимый объем памяти и, следовательно, максимально возможный размер программы и максимально возможный объем запоминаемых данных.

Количество адресов, обеспечиваемых шиной адреса, определяется как  $2^N$ , где  $N$  — количество разрядов. Например, 16-разрядная шина адреса обеспечивает 65 536 адресов.

Разрядность шины адреса обычно кратна 4 и может достигать 32 и даже 64. Шина адреса может быть однонаправленной (когда магистралью всегда управляет только процессор) или двунаправленной (когда процессор может временно передавать управление магистралью другому устройству, например контроллеру ПДП). Наиболее часто используются типы выходных каскадов с тремя состояниями или обычные ТТЛ (с двумя состояниями).

Как в шине данных, так и в шине адреса может использоваться положительная логика или отрицательная логика. При положительной логике высокий уровень напряжения соответствует логической единице на соответствующей линии связи, низкий — логическому нулю. При отрицательной логике — наоборот. В большинстве случаев уровни сигналов на шинах — ТТЛ.

Для снижения общего количества линий связи магистрали часто применяется мультиплексирование шин адреса и данных. То есть одни и те же линии связи используются в разные моменты времени для передачи как адреса, так и данных (в начале цикла — адрес, в конце цикла — данные). Для фиксации этих моментов (стробирования) служат специальные сигналы на шине управления.

**Шина данных** — это основная шина, которая используется для передачи информационных кодов между всеми устройствами микропроцессорной системы. Обычно в пересылке информации участвует процессор, который передает код данных в какое-то устройство или в ячейку памяти или же принимает код данных из какого-то устройства или из ячейки памяти. Но возможна также и передача информации между устройствами без участия процессора. Шина данных всегда двунаправленная.

Наиболее часто встречающийся тип выходного каскада для линий этой шины — выход с тремя состояниями.

Обычно шина данных имеет 8, 16, 32 или 64 разряда. Понятно, что за один цикл обмена по 64-разрядной шине может передаваться 8 байт информации, а по 8-разрядной — только один байт. Разрядность шины данных определяет и разрядность всей магистрали. Например, когда говорят о 32-разрядной системной магистрали, подразумевается, что она имеет 32-разрядную шину данных.

**Шина управления** в отличие от шины адреса и шины данных состоит из отдельных управляющих сигналов. Каждый из этих сигналов во время обмена информацией имеет свою функцию. Некоторые сигналы служат для стробирования передаваемых или принимаемых данных (то есть определяют моменты времени, когда информационный код выставлен на шину данных).

Шина управления — это вспомогательная шина, управляющие сигналы на которой определяют тип текущего цикла и фиксируют моменты времени, соответствующие разным частям или стадиям цикла. Кроме того, управляющие сигналы обеспечивают согласование работы процессора (или другого хозяина магистрали, задатчика, master) с работой памяти или устройства ввода/вывода (устройства-исполнителя, slave). Управляющие сигналы также обслуживают запрос и предоставление прерываний, запрос и предоставление прямого доступа.

Сигналы шины управления могут передаваться как в положительной логике (реже), так и в отрицательной логике (чаще). Линии шины управления могут быть как однонаправленными, так и двунаправленными. Типы выходных каскадов могут быть самыми разными: с двумя состояниями (для однонаправленных линий), с тремя состояниями (для двунаправленных линий), с открытым коллектором (для двунаправленных и мультиплексированных линий).

Самые главные управляющие сигналы — это стробы обмена, то есть сигналы, формируемые процессором и определяющие моменты времени, в которые производится пересылка данных по шине данных, обмен данными. Чаще всего в магистрали используются два различных строба обмена:

- Строб записи (вывода), который определяет момент времени, когда устройство-исполнитель может принимать данные, выставленные процессором на шину данных ;
- Строб чтения (ввода), который определяет момент времени, когда устройство-исполнитель должно выдать на шину данных код данных, который будет прочитан процессором.

При этом большое значение имеет то, как процессор заканчивает обмен в пределах цикла, в какой момент он снимает свой строб обмена. Возможны два пути решения (рис. 10):

- При синхронном обмене процессор заканчивает обмен данными самостоятельно, через раз и навсегда установленный временной интервал выдержки ( $t_{\text{выд}}$ ), то есть без учета интересов устройства-исполнителя;
- При асинхронном обмене процессор заканчивает обмен только тогда, когда устройство-исполнитель подтверждает выполнение операции специальным сигналом (так называемый режим handshake — рукопожатие).

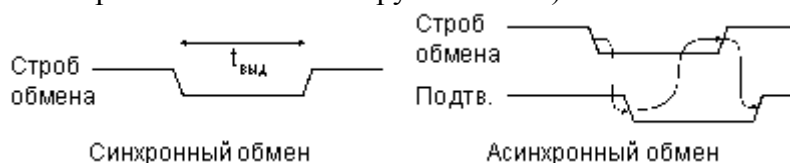


Рис. 10 Синхронный обмен и асинхронный обмен.

Достоинства синхронного обмена — более простой протокол обмена, меньшее количество управляющих сигналов. Недостатки — отсутствие гарантии, что исполнитель выполнил требуемую операцию, а также высокие требования к быстродействию исполнителя.

Достоинства асинхронного обмена — более надежная пересылка данных, возможность работы с самыми разными по быстродействию исполнителями. Недостаток — необходимость формирования сигнала подтверждения всеми исполнителями, то есть дополнительные аппаратные затраты.

*Какой тип обмена быстрее, синхронный или асинхронный? Ответ на этот вопрос неоднозначен. С одной стороны, при асинхронном обмене требуется какое-то время на выработку, передачу дополнительного сигнала и на его обработку процессором. С другой стороны, при синхронном обмене приходится искусственно увеличивать длительность строба обмена для соответствия требованиям большего числа исполнителей, чтобы они успевали обмениваться информацией в темпе процессора. Поэтому иногда в магистрали предусматривают возможность как синхронного, так и асинхронного обмена, причем синхронный обмен является основным и довольно быстрым, а асинхронный применяется только для медленных исполнителей.*

По используемому типу обмена магистрали микропроцессорных систем также делятся на синхронные и асинхронные.

*Необходимость стробирующих сигналов обусловлена следующими соображениями. В реальной схемотехнике невозможно обеспечить одинаковое время срабатывания устройств. Возможна ситуация, когда данные или адрес поступают на шину от нескольких логических элементов.*



Тогда в определенный момент времени на шине могут отказаться биты старого и нового сигнала. Что приведет к ошибке при передаче данных.

Стробирующий сигнал подается на шину управления тогда. Когда формирование сигнала на шинах адреса и данных завершено. Все устройства на шине по этому сигналу считывают корректные данные.

Например:

При обращении к ячейке оперативной памяти:

1. Выставляется адрес строки;
2. Подается стробирующий сигнал (RAS);
3. Выставляется адрес ячейки;
4. Подается стробирующий сигнал (CAS);

Для реализации такого поведения у логических элементов используется синхронизирующий вход.

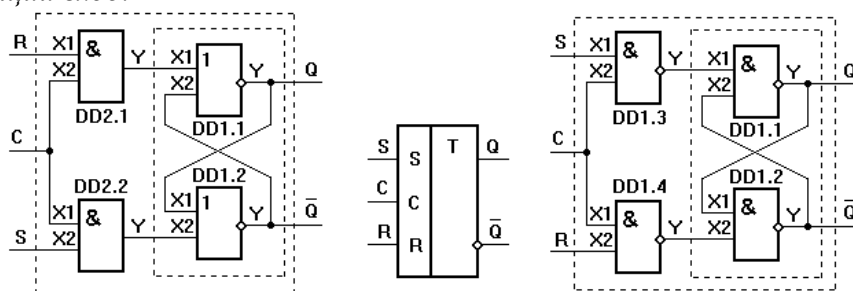


Рисунок 11 Функциональные схемы синхронного RS-триггера с прямыми входами, и условное обозначение этих триггеров на принципиальных схемах.

На рис. 11 показан пример синхронного RS триггера с статическим управлением. Вход C-синхронизирующий.

Другие управляющие сигналы могут использоваться для подтверждения приема данных, для сброса всех устройств в исходное состояние, для тактирования всех устройств и т.д. Линии шины управления могут быть однонаправленными или двунаправленными.

**Шина питания** предназначена не для пересылки информационных сигналов, а для питания системы. Она состоит из линий питания и общего провода. В микропроцессорной системе может быть один источник питания (чаще +5 В) или несколько источников питания (обычно еще -5 В, +12 В и -12 В). Каждому напряжению питания соответствует своя линия связи. Все устройства подключены к этим линиям параллельно.

## Работа микропроцессорной системы с шиной

Если в микропроцессорную систему надо ввести входной код (или входной сигнал), то процессор по шине адреса обращается к нужному устройству ввода/вывода и принимает по шине данных входную информацию.

Если из микропроцессорной системы надо вывести выходной код (или выходной сигнал), то процессор обращается по шине адреса к нужному устройству ввода/вывода и передает ему по шине данных выходную информацию.

Если информация должна пройти сложную многоступенчатую обработку, то процессор может хранить промежуточные результаты в системной оперативной памяти. Для обращения к любой ячейке памяти процессор выставляет ее адрес на шину адреса и передает в нее информационный код по шине данных или же принимает из нее информационный код по шине данных.

В памяти (оперативной и постоянной) находятся также и управляющие коды (команды выполняемой процессором программы), которые процессор также читает по шине данных с адресацией по шине адреса.

Постоянная память используется в основном для хранения программы начального

пуска микропроцессорной системы, которая выполняется каждый раз после включения питания. Информация в нее заносится изготовителем раз и навсегда.

Таким образом, в микропроцессорной системе все информационные коды и коды команд передаются по шинам последовательно, по очереди. Это определяет сравнительно невысокое быстродействие микропроцессорной системы. Оно ограничено обычно даже не быстродействием процессора (которое тоже очень важно) и не скоростью обмена по системной шине (магистралах), а именно последовательным характером передачи информации по системной шине (магистралах).

Важно учитывать, что устройства ввода/вывода чаще всего представляют собой устройства на "жесткой логике". На них может быть возложена часть функций, выполняемых микропроцессорной системой. Поэтому у разработчика всегда имеется возможность перераспределять функции системы между аппаратной и программной реализациями оптимальным образом.

Аппаратная реализация ускоряет выполнение функции, но имеет недостаточную гибкость.

Программная реализация значительно медленнее, но обеспечивает высокую гибкость. Аппаратная реализация функций увеличивает стоимость системы и ее энергопотребление, программная — не увеличивает. Чаще всего применяется комбинирование аппаратных и программных функций.

Задачи требующие высокой скорости выполнения, которые выполняются по одинаковым алгоритмам и не требуют гибкости настроек выполняются аппаратными средствами. Такими задачами обычно является обслуживание системной магистрали, низкоуровневое управление контроллерами.

Задачи, которые в силу сложности или необходимости регулярной адаптации, невозможно передать аппаратными средствами решаются программно.

Иногда устройства ввода/вывода имеют в своем составе процессор, то есть представляют собой небольшую специализированную микропроцессорную систему. Это позволяет переложить часть программных функций на устройства ввода/вывода, разгрузив центральный процессор системы. Ярким примером такого решения является делегирование задач по расчету 3D изображения видеоконтроллерам (видеокартам).

В настоящее время видеоконтроллерам успешно передают однообразные математические вычисления не связанные с расчетом изображения<sup>3</sup>. Например при кодировании и шифровании информации<sup>4</sup>.

## **Прохождение сигналов по магистрали**

При организации обмена по магистралям и шинам разработчику необходимо учитывать несколько важных моментов, связанных как с особенностью распространения сигналов по шинам, так и с самой природой шин. В противном случае микропроцессорная система может попросту не работать или работать неустойчиво, хотя вся логика цифровых устройств, входящих в систему, будет спроектирована безошибочно.

В случае, когда системная шина (магистраль) микропроцессорной системы является внешней, а не скрыта внутри микросхемы, необходимо учитывать особенности распространения сигналов по длинным линиям. Хотя в большинстве случаев длина магистрали не слишком велика, не превышает 1—2 десятков сантиметров, это все равно оказывает большое влияние на синхронизацию обмена.

На прохождение сигналов по магистрали влияют следующие факторы:

- конечная величина задержки распространения сигналов по линиям магистрали;
- различие задержек распространения сигналов по разным линиям шины;

3 <http://ru.wikipedia.org/wiki/CUDA>

4 [http://50btc.com/ru/article/bitcoin\\_mining\\_manual\\_dennis\\_lee](http://50btc.com/ru/article/bitcoin_mining_manual_dennis_lee) , раздел «GPU майнинг»

- неодновременное выставление сигналов на линии шины;
- искажение фронтов сигналов, проходящих по линиям магистрали;
- отражение сигналов от концов линий связи (рис. 23).



**Рис 23. Прохождение сигналов по шине.**

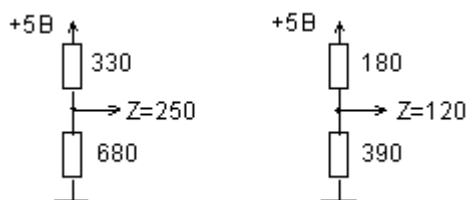
Для учета всех этих факторов разработчики стандартных магистралей обмена и стандартных протоколов обмена всегда закладывают необходимые задержки между сигналами, участвующими в обмене. Кроме того, задержки между сигналами выбираются таким образом, чтобы устройству, которому адресован тот или иной сигнал, имело достаточно времени для его обработки. Если разрабатывается новая магистраль, все это тоже надо учитывать.

Поэтому пытаться "модернизировать" какой-то стандартный протокол и ускорять обмен по магистрали путем уменьшения задержек, предусмотренных стандартом, очень опасно. Точно так же опасно, не изменяя протокола обмена, пытаться увеличить длину магистрали, увеличивая тем самым задержки распространения сигналов по линиям и шинам. Особенно чувствительны к такого рода "модернизациям" синхронные магистрали, в которых не предусмотрено обязательное подтверждение выполнения каждой операции.

Например, длительность фазы адреса в цикле обмена выбирается таким образом. В течение адресной фазы все сигналы всех разрядов кода адреса, пусть даже и сформированные процессором не одновременно, должны дойти до устройства-исполнителя по своим проводам шины. А устройство-исполнитель должно этот код адреса принять и обработать (то есть отличить свой адрес от чужого). Естественно, для гарантии в длительность адресной фазы еще добавляется небольшая дополнительная задержка.

Точно так же длительность фазы данных в цикле чтения должна выбираться такой, чтобы устройство-исполнитель успело получить строб чтения и выдать код читаемых данных на шину данных. Затем этот код должен успеть дойти до процессора и процессор должен успеть его прочесть. После чего процессор снимает сигнал строга чтения, этот задний фронт сигнала доходит с задержкой до устройства-исполнителя, которое также с задержкой снимает свой код данных. Аналогично и в цикле записи.

Для улучшения формы сигналов, распространяющихся по магистрали, иногда применяют оконечные согласователи (терминаторы) на концах линий магистрали. Особенно важно их применение в случае, когда допустимая длина магистрали превышает несколько метров. Например, в случае магистрали Q-bus применяются два типа согласователей: 120-омный и 250-омный (рис. 24).



**Рис. 24 Оконечные согласователи на магистрали Q-bus.**

Включение согласователей предъявляет дополнительные требования к нагрузочной способности передатчиков, работающих на линии магистрали. В магистрали ISA подобные

согласователи не используются, хотя к некоторым линиям присоединены резисторы, соединенные другим своим выводом с шиной питания (прежде всего это линии, тип выходного каскада для которых — ОК).

В любом случае выходные каскады передатчиков, работающих на линии магистрали, должны обеспечивать высокие выходные токи, так как к магистрали может подключаться несколько устройств, каждое из которых потребляет входной ток. Типичные величины требуемых выходных токов магистральных передатчиков находятся в пределах 20—30 мА. В то же время входные токи магистральных приемников должны быть малыми, чтобы не перегружать передатчики. Типичные величины допустимых входных токов магистральных приемников лежат в пределах 0,2—0,8 мА.

## Вопросы для контроля

### Обязательный уровень

1. Что понимают под архитектурой ЭВМ?
2. Назовите общие принципы построения ЭВМ.
3. По каким признакам определяется принадлежность систем к одному семейству?
4. Охарактеризуйте при знак информационной совместимости.
5. Охарактеризуйте при знак программной совместимости.
6. Охарактеризуйте при знак аппаратной совместимости.
7. Охарактеризуйте при знак конструктивной совместимости.
8. Охарактеризуйте при знак эксплуатационной совместимости.
9. Назовите основные положения выдвинутые фон Нейманом
10. Назовите функции основных элементов системы построенной по фон Неймановской архитектуре
11. Опишите классическую и шинную структуры связей
12. Перечислите недостатки шинной структуры связей
13. Что такое мультиплексированная линия? Двухнаправленная линия?
14. Опишите типичную структуру микропроцессорной системы. Перечислите ее компоненты и их роль.
15. Перечислите шины из которых обычно состоит системная магистраль.
16. Назовите основные параметры шин.
17. Что понимают под разрядностью шин?
18. Какова роль шины адреса? Может ли эта шина быть однонаправленной?
19. Какова роль шины данных? Может ли эта шина быть однонаправленной?
20. Какова роль шины управления? Может ли эта шина быть однонаправленной?
21. Какова роль шины питания?
22. Как выполняется ввод, вывод и многоступенчатая обработка данных?
23. Чем отличается аппаратная и программная реализация функций системы?
24. Что влияет на прохождение сигналов по магистрали?
25. Зачем используются терминаторы?
26. Перечислите типы микропроцессорных систем
27. Какие системы относятся к микроконтроллерам и микрокомпьютерам?
28. Какие системы относятся к компьютерам и контроллерам?

### Расширенный уровень

1. В каких направлениях ведется улучшение характеристик моделей одного семейства. Приведите примеры
2. Назовите направления совершенствования семейств ЭВМ. Приведите примеры
3. Назовите недостатки классической структуры связей и опишите как они решаются в шинной структуре
4. Какие выходные каскады цифровых систем могут использоваться в системах с шинной структурой? Опишите их
5. Какие выходные каскады позволяют объединять выводы микросхем в мультиплексированных и двухнаправленных линиях? Почему?
6. Какую характеристику микропроцессорной системы определяет разрядность шины адреса? Почему?
7. Какую характеристику микропроцессорной системы определяет разрядность шины данных? Почему?
8. Какие стробы обмена чаще всего используются в магистрали?
9. Охарактеризуйте синхронный и асинхронный обмен. Как тип обмена влияет на скорость работы магистрали?
10. Что такое асинхронные, синхронные, маскируемые и немаскируемые прерывания?
11. Какие трудности возникают при увеличении производительности магистрали?

### **Источники:**

Семейства ЭВМ и требования к ним – Url: <http://www.8086.ru/architecture/groups-of-computers.html>

Основы микропроцессорной техники. Новиков Юрий Витальевич, Скоробогатов Петр Константинович — Url: <http://www.intuit.ru/studies/courses/3/3/info>