

# Микропроцессорная техника в мехатронике и робототехнике



Кафедра «Робототехника и мехатроника»

## Лекционный курс

Автор

Герасимов В.А.

## **Аннотация**

В компактной форме приводится информация, необходимая для изучения дисциплин «Микропроцессорная техника» и «Микропроцессорная техника в мехатронике и робототехнике».

Предназначены для студентов 3-го курса подготовки бакалавров направлений 09.03.01 «Информатика и вычислительная техника» и 15.03.06 «Мехатроника и робототехника».

## **Автор**

**Герасимов Владимир Анатольевич –**

**КАНДИДАТ ТЕХНИЧЕСКИХ НАУК, ДОЦЕНТ, ПРОФЕССОР**

## ОГЛАВЛЕНИЕ

1. Основные сведения об электронных системах .....	4
2. Структурные схемы микропроцессоров и микроконтроллеров .....	8
3. Архитектура микропроцессоров и микроконтроллеров .....	12
4. Обмен информацией в микропроцессорах и микроконтроллерах.....	18
5. Системы команд микропроцессоров и микроконтроллеров .....	23
6. Перспективы применения микропроцессорной техники в системах управления.....	28
Литература .....	30

# 1. Основные сведения об электронных системах

## Термины и определения

Развитие мехатроники и робототехники невозможно без применения современных электронных систем.

Под электронной системой понимают любой электронный узел, блок, прибор или комплекс, производящий обработку и хранение информации. Электронная система может быть реализована в виде электронной схемы аппаратно выполняющей заданный схмотехнически алгоритм обработки информации. Такую электронную систему часто называют системой на «жесткой логике». Другой вариант электронной системы – микропроцессорная система.

Электронная система имеет входы и выходы для информационных сигналов (рис. 1.1). В качестве информационных сигналов используются аналоговые, импульсные и цифровые сигналы. Аналоговые сигналы передают информацию своей формой, импульсные – временными параметрами импульсной последовательности, цифровые – специально разработанными кодами. В аналоговых сигналах усиление и ослабление сигнала без искажения формы информацию не изменяет. Для передачи цифровой информации в качестве носителя используют импульсные сигналы, при этом их временные параметры не учитываются /1/.



Рис. 1.1. Электронная система

Основными терминами и определениями используемыми для описания электронных систем являются:

- задача — набор функций, выполнение которых требуется от электронной системы;

## Микропроцессорная техника

- быстроедействие — показатель скорости выполнения электронной системой ее функций;

- гибкость — это способность системы подстраиваться под различные задачи;

- избыточность — это показатель степени соответствия возможностей системы решаемой данной системой задаче;

- интерфейс — соглашение об обмене информацией, правила обмена информацией, подразумевающие электрическую, логическую и конструктивную совместимость устройств, участвующих в обмене.

Характерная особенность традиционной цифровой системы на «жесткой логике» состоит в том, что алгоритмы обработки и хранения информации в ней жестко связаны со схемотехникой системы [3]. То есть изменение этих алгоритмов возможно только путем изменения структуры системы, замены электронных узлов, входящих в систему, и/или связей между ними. Например, если потребуется дополнительная операция суммирования, то необходимо добавить в структуру системы лишний сумматор. Естественно, это практически невозможно сделать в процессе эксплуатации, обязательно нужен новый производственный цикл проектирования, изготовления, отладки всей системы. Системы на «жесткой логике» хороши там, где решаемая задача не меняется длительное время, где требуется самое высокое быстроедействие, где алгоритмы обработки информации предельно просты.

Достоинства систем на «жесткой логике»:

- нет аппаратной избыточности;

- максимально высокое быстроедействие, так как скорость выполнения алгоритмов обработки информации определяется в ней только быстроедействием отдельных логических элементов и выбранной схемой путей прохождения информации;

- высокая надежность из-за отсутствия избыточности;

- малое энергопотребление, при использовании современной технологической базы.

Недостатки систем на «жесткой логике»:

- отсутствие гибкости;

- сложность проектирования;

- высокая стоимость проектирования;

### Микропроцессорная техника

- сложность и длительность изготовления и отладки.

Микропроцессорная система (МС) может рассматриваться как частный случай электронной системы, предназначенной для обработки входных сигналов и выдачи выходных сигналов. Микропроцессорные системы являются универсальными электронными системами. За последние десятилетия быстродействие микропроцессорных систем значительно выросло. Они относятся к разряду цифровых устройств. Для обработки входных аналоговых сигналов они преобразуются в цифровые с помощью аналогово-цифровых преобразователей (АЦП, ADC), а выходные аналоговые сигналы формируются из цифровых – обратным преобразованием с помощью цифро-аналоговых преобразователей (ЦАП, DAC). Обработка и хранение информации в микропроцессорной системе производится в цифровом виде.

МС легко адаптировать под любую задачу, перестраивая её с одного алгоритма работы на другой без изменения её схемотехники. Задают тот или иной алгоритм путем ввода в систему некой дополнительной управляющей информации – программы работы системы (рис. 1.2). Тогда система станет универсальной, или программируемой, не жесткой, а гибкой.

Достоинства микропроцессорных систем:

- большая гибкость;
- практически любая сложность решаемой МС задачи;
- малое время проектирования и изготовления.

Недостатки микропроцессорных систем:

- потенциально меньшее быстродействие;
- большая избыточность и как следствие – ниже надежность, большее потребление электроэнергии;
- высокая стоимость изделия.

Универсальные микропроцессорные системы хороши там, где часто меняются решаемые задачи, где высокое быстродействие не слишком важно, где требуются сложные алгоритмы обработки информации.

## Микропроцессорная техника



Рис. 1.2. Программируемая (микропроцессорная) электронная система

Прогресс в технологии изготовления и большой объем выпуска микросхем для микропроцессорных систем привел к резкому снижению их стоимости. В результате область применения систем на «жесткой логике» резко сузилась. Развитие получили программируемые системы на базе микроконтроллеров, предназначенные для решения одной задачи или нескольких близких задач. Они удачно совмещают в себе как достоинства систем на «жесткой логике», так и программируемых систем, обеспечивая сочетание достаточно высокого быстродействия и необходимой гибкости.

## 2. Структурные схемы микропроцессоров и микроконтроллеров

Ядром любой микропроцессорной системы является микропроцессор или микроконтроллер, так как именно микропроцессор — это тот узел, блок, который производит всю обработку информации внутри микропроцессорной системы. Остальные узлы выполняют вспомогательные функции: хранение информации (в том числе и управляющей информации, то есть программы), связи с внешними устройствами, связи с пользователем и т. д. Процессор выполняет арифметические функции (сложение, умножение и т. д.), логические функции (сдвиг, сравнение, маскирование кодов и т. д.), временное хранение кодов (во внутренних регистрах), пересылку кодов между узлами микропроцессорной системы и многое другое. Количество таких элементарных операций, выполняемых процессором, может достигать нескольких сотен. Но при этом надо учитывать, что все свои операции процессор выполняет последовательно, то есть одну за другой, по очереди. Существуют процессоры с параллельным выполнением некоторых операций, встречаются также микропроцессорные системы, в которых несколько процессоров работают над одной задачей параллельно. С одной стороны, последовательное выполнение операций — несомненное достоинство, так как позволяет с помощью всего одного процессора выполнять любые, самые сложные алгоритмы обработки информации. Но, с другой стороны, последовательное выполнение операций приводит к тому, что время выполнения алгоритма зависит от его сложности. Простые алгоритмы выполняются быстрее сложных. То есть микропроцессорная система способна сделать все, но все информационные потоки приходится пропускать через один-единственный узел — микропроцессор.

Выполнение операций микропроцессором определяется управляющей информацией, программой. Программа представляет собой набор команд (инструкций), то есть цифровых кодов, декодировав которые, процессор определяет, что ему надо делать.

Все команды, выполняемые процессором, образуют систему команд процессора. Структура и объем системы команд процессора определяют его быстродействие, гибкость, удобство использования. Всего команд у процессора может быть от нескольких десятков до нескольких сотен. Система команд может быть расчи-

Микропроцессорная техника

тана на узкий круг решаемых задач (у специализированных процессоров) или на максимально широкий круг задач (у универсальных процессоров). Коды команд могут иметь различное количество разрядов (занимать от одного до нескольких байт). Каждая команда имеет свое время выполнения, поэтому время выполнения всей программы зависит не только от количества команд в программе, но и от того, какие именно команды используются.

Таким образом, любой процессор должен содержать операционную и управляющие части.

В операционную часть входят арифметико - логическое устройство и мультиплексоры, буферы, регистры и другие узлы, в том числе — специализированный регистр управления и состояния (часто называемый регистром признаков или слова состояния процессора PSW).

Управляющая часть включает устройство микропрограммного управления с дешифратором кода операции и схемы управления выборкой команд, прерываниями и прямым доступом к памяти. Работа всех узлов синхронизируется общим внешним тактовым сигналом (рис. 2.1).

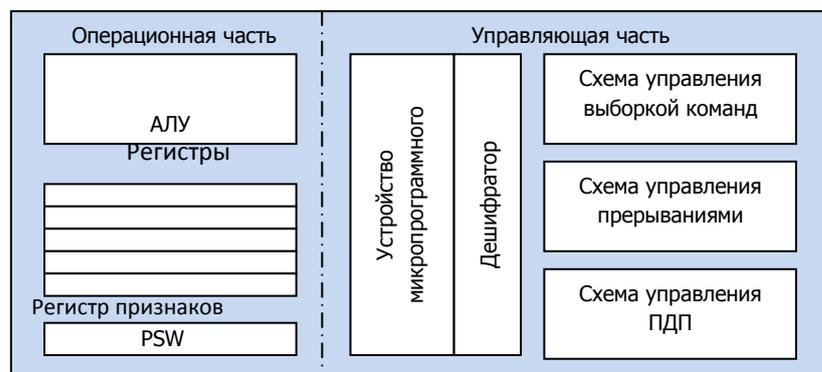


Рис. 2.1. Структура простейшего универсального процессора

Для разработчика микропроцессорных систем информация о тонкостях внутренней структуры процессора не слишком важна. Разработчик должен рассматривать процессор как «черный ящик», который в ответ на входные и управляющие коды производит ту или иную операцию и выдает выходные сигналы. Разработчику необходимо знать систему команд, режимы работы процессора, а также правила взаимодействия процессора с внешними устройствами, т. е. протоколы обмена информацией.

## Микропроцессорная техника

Большой чем микропроцессоры функциональностью обладают микроконтроллеры (МК) /5,6/. В настоящее время выпускается целый ряд типов МК. Все эти устройства можно условно разделить на три основных класса:

- 8-разрядные МК для встраиваемых приложений;
- 16- и 32-разрядные МК для систем верхнего уровня управления;
- цифровые сигнальные процессоры (DSP) для работы с аналоговыми сигналами.

МК прошли в своем развитии путь от простейших приборов с относительно слаборазвитой периферией до современных многофункциональных контроллеров, обеспечивающих реализацию сложных алгоритмов управления в реальном масштабе времени и представляющих серьезную конкуренцию универсальным процессорам. Росту популярности МК способствует постоянное расширение номенклатуры и функциональности изделий, выпускаемых такими известными фирмами, как Motorola, Microchip, Intel, Zilog, Atmel и многими другими.

Современные МК представляют собой микропроцессорную систему из микропроцессора, контроллеров ввода-вывода, объединенных внутренними шинами, и схем синхронизации и контроля работы. МК обладают рядом отличительных признаков. Основные из них:

- модульная организация, при которой на базе одного процессорного ядра (ЦПУ) проектируется ряд (линейка) МК, различающихся объемом и типом памяти программ, объемом памяти данных, набором периферийных модулей, частотой синхронизации;
- использование закрытой архитектуры МК, которая характеризуется отсутствием линий магистралей адреса и данных на выводах корпуса МК. Таким образом, МК представляет собой законченную систему обработки данных, наращивание возможностей которой с использованием параллельных магистралей адреса и данных не предполагается;
- использование типовых функциональных периферийных модулей (таймеры, процессоры событий, контроллеры последовательных интерфейсов, аналого-цифровые преобразователи и др.), имеющих незначительные отличия в алгоритмах работы в МК различных производителей;
- расширение числа режимов работы периферийных модулей, которые задаются в процессе инициализации регистров специальных функций МК.

## Микропроцессорная техника

При модульном принципе построения все МК одного семейства содержат процессорное ядро, одинаковое для всех МК данного семейства, и изменяемый функциональный блок, который отличает МК разных моделей. Структура типичного модульного МК приведена на рис. 2.2.

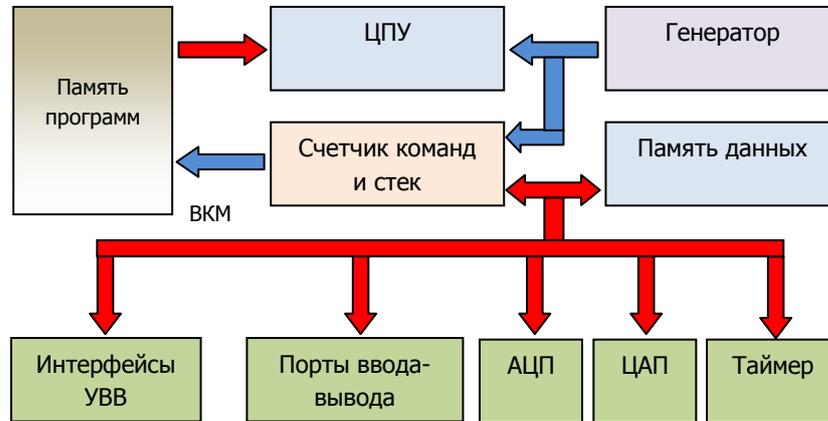


Рис. 2.2. Структура типичного микроконтроллера

Процессорное ядро включает в себя:

- центральный процессор (ЦПУ);
- внутреннюю контроллерную магистраль (ВКМ) в составе шин адреса, данных и управления;
- схему синхронизации МК (Генератор);
- схему управления режимами работы МК, включая поддержку режимов пониженного энергопотребления, начального запуска (сброса) и т. д. (счетчик команд и стек).

Изменяемый функциональный блок включает в себя модули памяти различного типа и объема (памяти программ и данных), порты ввода/вывода, модули АЦП и ЦАП, таймеры, различные контроллеры стандартных интерфейсов. В относительно простых МК модуль обработки прерываний входит в состав процессорного ядра. В более сложных МК он представляет собой отдельный модуль с развитыми возможностями. В состав изменяемого функционального блока могут входить дополнительные модули. Каждый модуль проектируется для работы в составе МК с учетом протокола ВКМ. Данный подход позволяет создавать разнообразные по структуре МК в пределах одного семейства.

### 3. Архитектура микропроцессоров и микроконтроллеров

В последние годы широко используется несколько модификаций классической структуры процессоров.

С точки зрения системы команд и способов адресации операндов в процессорах реализуется один из двух принципов построения:

- процессоры с CISC-архитектурой, реализующие так называемую полную систему команд (Complicated Instruction Set Computer);
- процессоры с RISC-архитектурой, реализующие сокращенную систему команд (Reduced Instruction Set Computer).

CISC-процессоры выполняют большой набор команд с развитыми возможностями адресации, давая разработчику возможность выбрать наиболее подходящую команду для выполнения необходимой операции. Процессор с CISC-архитектурой может иметь однобайтовый, двухбайтовый и трехбайтовый (редко четырехбайтовый) формат команд. При этом система команд, как правило, не ортогональна, то есть не все команды могут использовать любой из способов адресации применительно к любому из регистров процессора. Выборка команды на исполнение осуществляется побайтно в течение нескольких циклов работы процессора.

В связи с успехами микроэлектроники появилась возможность построения RISC-компьютеров (Reduced Instruction Set Computing), т. е. микропроцессорных систем с сокращенным набором команд.

Поэтому в RISC архитектуре основу системы команд составляют наиболее употребительные, «короткие» операции типа алгебраического сложения. Сложные операции выполняются как подпрограммы, состоящие из простых операций. Это позволяет значительно упростить внутреннюю структуру процессора, уменьшить фазы дробления конвейерной обработки и увеличить частоту работы конвейера. Но здесь необходимо отметить, что за эффект приходится расплачиваться усложнением процедур обмена данными между регистрами, кэш-памятью и оперативной памятью.

В процессорах с RISC-архитектурой набор исполняемых команд сокращен до минимума. Для реализации более сложных операций приходится комбинировать

### Микропроцессорная техника

команды. При этом все команды имеют формат фиксированной длины (например, 12, 14 или 16 бит), выборка команды из памяти и ее исполнение осуществляется за один цикл (такт) синхронизации. Система команд RISC-процессора предполагает возможность равноправного использования всех регистров процессора. Это обеспечивает дополнительную гибкость при выполнении ряда операций.

Вопрос о производительности процессоров с архитектурой RISC и CISC сложен и неоднозначен.

Во-первых, оценка производительности по времени выполнения команд различных систем (RISC и CISC) не совсем корректна. Обычно производительность МП и МК принято оценивать числом операций пересылки «регистр-регистр», которые могут быть выполнены в течение одной секунды. В системах с CISC-процессором время выполнения операции «регистр-регистр» составляет от 1 до 3 циклов, что, казалось бы, уступает производительности систем с RISC-процессором. Однако стремление к сокращению формата команд при сохранении ортогональности системы команд RISC-процессора приводит к вынужденному ограничению числа доступных в одной команде регистров. Так, например, системой команд МК PIC16 предусмотрена возможность пересылки результата операции только в один из двух регистров — регистр-источник операнда  $f$  или рабочий регистр  $W$ . Таким образом, операция пересылки содержимого одного из доступных регистров в другой (не источник операнда и не рабочий) потребует использования двух команд. Такая необходимость часто возникает при пересылке содержимого одного из регистров общего назначения (РОН) в один из портов МК. В то же время, в системе команд большинства CISC-процессоров присутствуют команды пересылки содержимого РОН в один из портов ввода/вывода. То есть более сложная система команд иногда позволяет реализовать более эффективный способ выполнения операции.

Во-вторых, оценка производительности по скорости пересылки «регистр-регистр» не учитывает особенностей конкретного реализуемого алгоритма управления. Так, при разработке быстродействующих устройств автоматизированного управления основное внимание следует уделять времени выполнения операций умножения и деления при реализации уравнений различных передаточных функций. А при реализации пульта дистанционного

## Микропроцессорная техника

управления бытовой техникой следует оценивать время выполнения логических функций, которые используются при опросе клавиатуры и генерации последовательной кодовой посылки управления. Поэтому в критических ситуациях, требующих высокого быстродействия, следует оценивать производительность на множестве тех операций, которые преимущественно используются в алгоритме управления и имеют ограничения по времени выполнения.

В-третьих, необходимо еще учитывать, что указанные в справочных данных частоты синхронизации обычно соответствуют или пропорциональны частоте кварцевого резонатора, в то время как длительность цикла центрального процессора определяется частотой обмена по системной магистрали. Соотношение этих частот индивидуально для каждого устройства должно быть принято в расчет при сравнении производительности различных вычислительных систем.

Другой подход к архитектуре вычислительных систем с точки зрения использования памяти в процессорах реализует один из двух принципов построения:

- архитектура с общей памятью для размещения команд и данных;
- архитектура с разделенной памятью команд и данных.

Для универсальных процессоров типичным является использование архитектуры с общей, единой шиной для данных и команд (одношинная, или принстонская, фон-неймановская архитектура). Соответственно, в составе системы в этом случае используется одна общая память, как для данных, так и для команд (рис. 3.1).

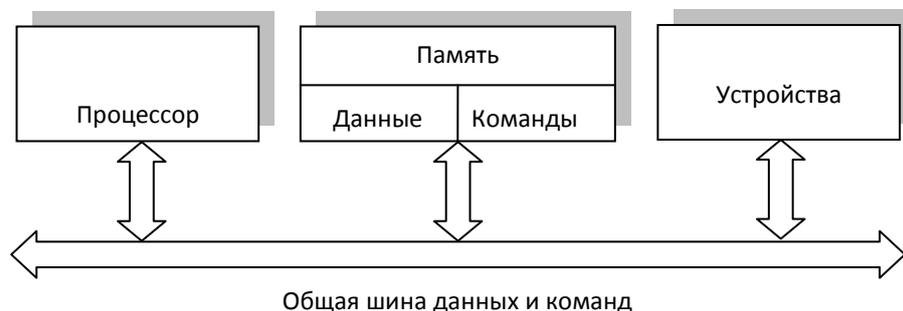


Рис. 3.1. Архитектура с общей шиной данных и команд

Другой тип архитектуры микропроцессорной системы — это архитектура с отдельными шинами данных и команд (двухшинная, или гарвардская,

Микропроцессорная техника

архитектура). Эта архитектура предполагает наличие в системе отдельной памяти для данных и отдельной памяти для команд (рис. 3.2). Обмен процессора с каждым из двух типов памяти происходит по своей шине.

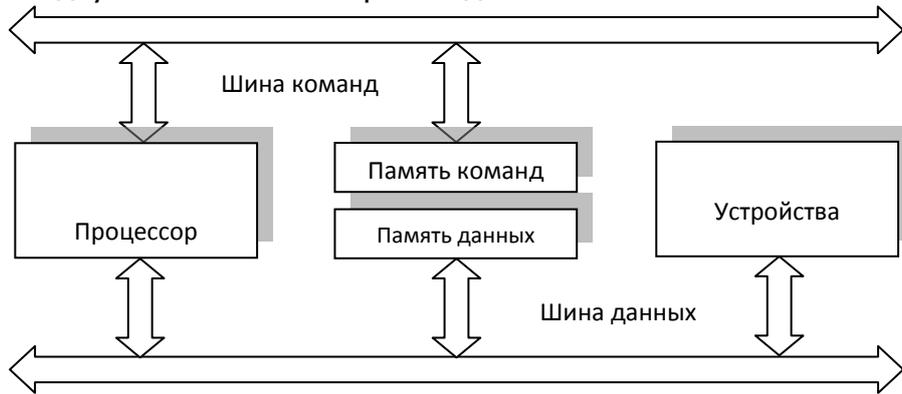


Рис. 3.2. Архитектура с отдельными шинами данных и команд

Архитектура с общей шиной распространена гораздо больше, она применяется, например, в персональных компьютерах и в сложных микрокомпьютерах. Архитектура с отдельными шинами применяется в основном в однокристальных микроконтроллерах.

Рассмотрим некоторые достоинства и недостатки обеих архитектурных решений.

Архитектура с общей шиной (принстонская, фон-неймановская) проще, она не требует от процессора одновременного обслуживания двух шин, контроля обмена по двум шинам сразу. Наличие единой памяти данных и команд позволяет гибко распределять ее объем между кодами данных и команд. Например, в некоторых случаях нужна большая и сложная программа, а данных в памяти надо хранить не слишком много. В других случаях, наоборот, программа требуется простая, но необходимы большие объемы хранимых данных. Перераспределение памяти производится оперативно, единственное условие — чтобы команды и данные в сумме помещались в памяти системы. Как правило, в системах с такой архитектурой память бывает довольно большого объема (до десятков гигабайт). Это позволяет решать самые сложные задачи.

Архитектура с отдельными шинами данных и команд сложнее, она заставляет процессор работать одновременно с двумя потоками кодов, обслуживать обмен по двум шинам одновременно. Программа может размещаться только в памяти команд, данные — только в памяти данных. Такая узкая

### Микропроцессорная техника

специализация ограничивает круг задач, решаемых системой, так как не дает возможности гибкого перераспределения памяти. Память данных и память команд в этом случае имеют разный объем (память команд значительно больше памяти данных), поэтому применение систем с данной архитектурой ограничивается обычно не слишком сложными задачами.

Преимущество гарвардской архитектуры в первую очередь, в быстродействии.

Дело в том, что при единственной шине команд и данных процессор вынужден по одной этой шине принимать данные (из памяти или устройства ввода/вывода) и передавать данные (в память или в устройство ввода/вывода), а также читать команды из памяти. Естественно, одновременно эти пересылки кодов по магистрали происходить не могут, они должны производиться по очереди. Современные процессоры способны совместить во времени выполнение команд и проведение циклов обмена по системной шине. Использование конвейерных технологий и быстрой кэш-памяти позволяет им ускорить процесс взаимодействия со сравнительно медленной системной памятью. Повышение тактовой частоты и совершенствование структуры процессоров дают возможность сократить время выполнения команд. Но дальнейшее увеличение быстродействия системы возможно только при совмещении пересылки данных и чтения команд, то есть при переходе к архитектуре с двумя шинами.

В случае двухшинной архитектуры обмен по обеим шинам может быть независимым, параллельным во времени. Соответственно, структуры шин (количество разрядов кода адреса и кода данных, порядок и скорость обмена информацией и т.д.) могут быть выбраны оптимально для той задачи, которая решается каждой шиной. Поэтому при прочих равных условиях переход на двухшинную архитектуру ускоряет работу микропроцессорной системы, хотя и требует дополнительных затрат на аппаратуру, усложнения структуры процессора. Память данных в этом случае имеет свое распределение адресов, а память команд — свое.

Проще всего преимущества двухшинной архитектуры реализуются внутри одной микросхемы. В этом случае можно также существенно уменьшить влияние недостатков этой архитектуры. Поэтому основное ее применение — в

### Микропроцессорная техника

микроконтроллерах, от которых не требуется решения слишком сложных задач, но зато необходимо максимальное быстродействие при заданной тактовой частоте.

Таким образом, сложившаяся практика проектирования микропроцессорных систем показывает, что для решения задач управления на верхнем уровне используются универсальные микропроцессоры с принстонской, CISC архитектурой, а на нижнем – микроконтроллеры с гарвардской, и предпочтительно RISC архитектурой.

## 4. Обмен информацией в микропроцессорах и микроконтроллерах

Обмен информацией в микропроцессорных системах происходит в циклах обмена информацией /4,8/. Под циклом обмена информацией понимается временной интервал, в течение которого происходит выполнение одной элементарной операции обмена по шине. Например, пересылка данных из процессора в память или же пересылка данных из устройства ввода/вывода в процессор. В пределах одного цикла также может передаваться и несколько кодов данных.

Циклы обмена информацией делятся на два основных типа:

- Цикл записи (вывода), в котором процессор записывает (выводит) информацию;
- Цикл чтения (ввода), в котором процессор читает (вводит) информацию.

Во время каждого цикла устройства, участвующие в обмене информацией, передают друг другу информационные и управляющие сигналы в строго установленном порядке или, как еще говорят, в соответствии с принятым протоколом обмена информацией.

Длительность цикла обмена может быть постоянной или переменной, но она всегда включает в себя несколько периодов сигнала тактовой частоты системы. То есть даже в идеальном случае частота чтения информации процессором и частота записи информации оказываются в несколько раз меньше тактовой частоты системы.

Обмен информацией происходит с использованием системной магистрали (системной шины, интерфейса) микропроцессорной системы. Системный интерфейс содержит три основные информационные шины: адреса, данных и управления.

Шина данных — это основная шина по которой передается информация. Количество ее разрядов (линий связи) определяет скорость и эффективность обмена, а также максимально возможное количество команд.

Шина данных всегда двунаправленная, так как предполагает передачу информации в обоих направлениях. Наиболее часто встречающийся тип выходного каскада для линий этой шины — выход с тремя состояниями.

## Микропроцессорная техника

Обычно шина данных имеет 8, 16, 32 или 64 разряда. 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-разрядном системном интерфейсе, подразумевается, что он имеет 32-разрядную шину данных.

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 65536 адресов. Разрядность шины адреса обычно кратна 4 и достигает 64 разрядов.

Шина адреса может быть однонаправленной (когда магистралью всегда управляет только процессор) или двунаправленной (когда процессор может временно передавать управление магистралью другому устройству). Наиболее часто используются типы выходных каскадов с тремя состояниями или обычные.

Для снижения общего количества линий связи процессора 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 (сначала — адрес, затем — данные). Для фиксации этих моментов (стробирования) служат специальные сигналы на шине управления. Мультиплексированная шина адреса/данных обеспечивает меньшую скорость обмена, требует более длительного цикла обмена.

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 с работой памяти или устройства ввода/вывода. Управляющие сигналы также обслуживают запрос и предоставление прерываний, запрос и предоставление прямого доступа к памяти.

Линии шины управления могут быть как однонаправленными, так и двунаправленными.

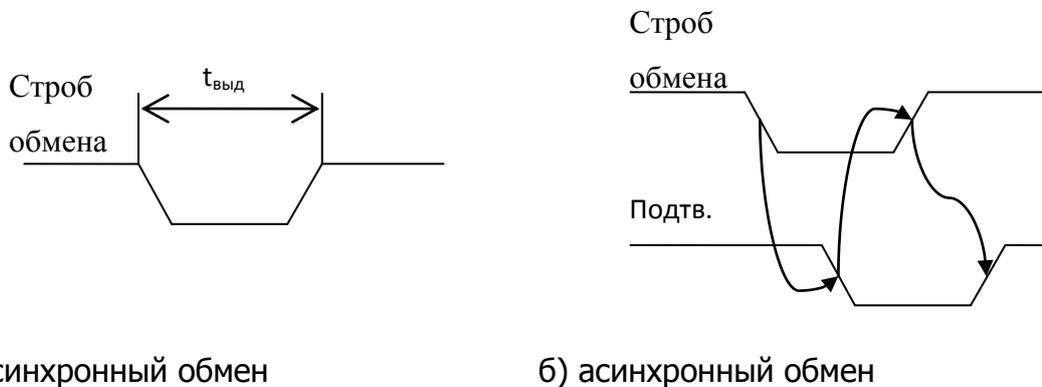
Основные управляющие сигналы — это стробы обмена, то есть сигналы, формируемые процессором и определяющие моменты времени, в которые

Микропроцессорная техника

производится пересылка данных по шине данных, обмен данными. Чаще всего в магистрали используются два различных строга 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. 4.1):



а) синхронный обмен

б) асинхронный обмен

Рис. 4.1. Синхронный обмен и асинхронный обмен

- при *синхронном* обмене процессор заканчивает обмен данными самостоятельно, через раз и навсегда установленный временной интервал выдержки ( $t_{\text{выд}}$ ), то есть без учета реального быстродействия остальных устройств участвующих в обмене;

- при *асинхронном* обмене процессор заканчивает обмен только тогда, когда устройство-исполнитель подтверждает выполнение операции специальным сигналом ответным сигналом /7/.

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.

## Микропроцессорная техника

Недостаток — необходимость формирования сигнала подтверждения всеми исполнителями, то есть дополнительные аппаратные затраты.

Синхронный и асинхронный типы обмена обеспечивают разную производительность работы микропроцессорной системы. 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 строка 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.

Обмен информацией в контроллерах устроен принципиально по-другому без использования внешнего системного интерфейса. Каждый МК имеет некоторое количество линий ввода/вывода, которые объединены в многоразрядные (чаще 8-разрядные) параллельные порты ввода/вывода. В памяти МК каждому порту ввода/вывода соответствует свой адрес регистра данных. Обращение к регистру данных порта ввода/вывода производится теми же командами, что и обращение к памяти данных. Кроме того, во многих МК отдельные разряды портов могут быть опрошены или установлены командами битового процессора.

В зависимости от реализуемых функций различают следующие типы параллельных портов:

- однонаправленные порты, предназначенные только для ввода или только для вывода информации;
- двунаправленные порты, направление передачи которых (ввод или вывод) определяется в процессе инициализации МК;
- порты с альтернативной функцией (мультиплексированные порты). Отдельные линии этих портов используются совместно со встроенными периферийными устройствами МК, такими как таймеры, АЦП, контроллеры последовательных интерфейсов;
- порты с программно управляемой схмотехникой входного/выходного буфера.

### Микропроцессорная техника

Порты выполняют роль устройств временного согласования функционирования МК и объекта управления, которые в общем случае работают асинхронно. Различают три типа алгоритмов обмена информацией между МК и внешним устройством через параллельные порты ввода/вывода:

- режим простого программного ввода/вывода;
- режим ввода/вывода со стробированием;
- режим ввода/вывода с полным набором сигналов подтверждения обмена.

В современных МК, как правило, обеспечивается индивидуальный доступ к портам, что позволяет использовать каждую линию независимо в режиме ввода или вывода.

Необходимо обратить особое внимание на то, что при вводе данных с порта считывается значение сигнала, поступающее на внешний вывод. При выводе информации на порт поступает содержимое регистра данных порта ассоциированное с одним из регистров оперативной памяти.

## 5. Системы команд микропроцессоров и микроконтроллеров

Система команд микропроцессорного устройства служит для связи между микропроцессором, аппаратурой и программным обеспечением и представляет ту часть системы, которая видна программисту /2/.

Все команды процессора можно разделить на несколько основных групп команд. Не смотря на большое число разновидностей МП и МК, на самом низком уровне системы их команд имеют много общего. Любая микропроцессорная система содержит следующие группы команд:

- команды передачи данных (копирования), копирующие информацию из одного места в другое;
- арифметические команды, производящие арифметические преобразования операндов;
- логические команды, позволяющие компьютеру производить анализ получаемой информации. Примерами могут служить сравнение, логические операции И, ИЛИ, НЕ, а так же анализ отдельных битов кода, их сброс и установка. Сдвиги двоичного кода влево и вправо. Операции сдвига используются, например, при выполнении умножения и деления чисел;
- команды ввода и вывода информации для обмена с внешними устройствами;
- команды управления, к которым следует отнести все виды переходов. Сюда же включают операции по управлению процессором.

Команды процессора обычно включают две части – операционную и адресную. Операционная часть (её называют кодом операции – КОП) указывает, какое действие необходимо выполнить с информацией. Адресная часть (адрес) указывает, где хранится используемая в операции информация и куда поместить результат. У некоторых команд управления работой процессора адресная часть может отсутствовать. Операционная часть команды имеется всегда. По количеству адресов, записываемых в команде, команды делятся на одно-, двух-, трех- и четырехадресные. Существует связь таких параметров процессора, как длина адресного пространства, адресность, разрядность. Увеличение разрядности позволяет увеличить адресность команды и длину адреса (т. е. объем памяти,

Микропроцессорная техника

доступной данной команде). Увеличение адресности, в свою очередь, приводит к повышению быстродействия обработки (за счет снижения числа требуемых команд).

Наибольшее количество адресов было в четырехадресных ЭВМ. В командах принята следующая последовательность указания частей – код операции – КОП, адрес первого операнда – A1, адрес второго операнда – A2, адрес результата – A3 и адрес следующей команды – A4. Такие команды применялись в первых ЭВМ (рис 5.1).

КОП – код операции	A1 – адрес первого операнда	A2 – адрес второго операнда	A3 – адрес результата	A4 – адрес следующей команды
--------------------------	-----------------------------------	-----------------------------------	--------------------------	------------------------------------

Рис. 5.1. Типовая структура четырехадресной команды

В микропроцессорах большое количество адресов снижает производительность. Поэтому, сначала отказались от использования адреса следующей команды, заменив его счетчиком команд. Затем результат начали помещать по адресу первого операнда. Это привело к настоящему времени к использованию двух-, одно- и безадресных команд (рис 5.2).

КОП – код операции	A1 – адрес первого операнда	A2 – адрес второго операнда
--------------------------	-----------------------------------	-----------------------------------

а)

КОП – код операции	A1 – адрес первого операнда
--------------------------	-----------------------------------

б)

КОП – код операции
--------------------------

в)

а) двухадресная команда; б) одноадресная команда; в) безадресная команда.

Рис 5.2. Типовая структура современных команд

Микропроцессорная техника

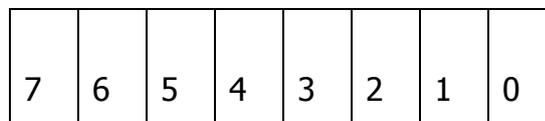
Таким образом, программирование в машинных командах требует знания системы команд конкретной ЭВМ и их адресности. При этом реализация даже довольно несложных вычислений требует разложения их на простые операции, что значительно увеличивает общий объем программы и затрудняет ее чтение и отладку.

Рассмотрим особенности команд используемых в универсальных микропроцессорах и микроконтроллерах.

Например, система команд одного из первых универсальных микропроцессоров i8080 состоит из 78 базовых команд, которые можно разделить на пять групп:

- команды передачи данных — используются для передачи данных из регистра в регистр, из памяти в регистр, из регистра в память;
- арифметические команды — используются для сложения, вычитания, инкремента или декремента содержимого регистров или ячейки памяти;
- логические команды: И, ИЛИ, исключающее ИЛИ, сравнение, сдвиги;
- команды переходов — используются для условных и безусловных переходов, вызова подпрограмм и возврата из них;
- команды управления, ввода/вывода и работы со стеком — используются для управления прерыванием, регистром признаков, ввода и вывода информации.

В микропроцессоре принят формат информационного слова, представляющего собой 8-разрядное двоичное слово (байт). Формат информационного слова (данных):

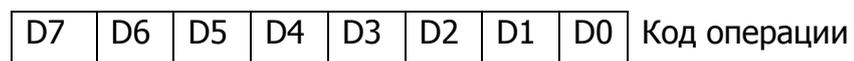


где D7 — старший разряд слова, D0 — младший разряд.

Формат команды зависит от типа операции и может быть одно-, двух-, или трехбайтовым. Байты двух- и трехбайтовых команд должны храниться в ячейках памяти, следующих одна за другой. Адрес первого байта всегда является адресом кода операции.

Формат команд процессора:

Однобайтовая команда



Микропроцессорная техника

Байт В1	Двухбайтовая команда								
Байт В1	D7	D6	D5	D4	D3	D2	D1	D0	Код операции
Байт В2	D7	D6	D5	D4	D3	D2	D1	D0	Данные или адрес
	Трехбайтовая команда								
Байт В1	D7	D6	D5	D4	D3	D2	D1	D0	Код операции
Байт В2	D7	D6	D5	D4	D3	D2	D1	D0	Данные или адрес
Байт В3	D7	D6	D5	D4	D3	D2	D1	D0	Данные или адрес

Как видно из формата команды универсального процессора, команды имеют фиксированный размер кода операции и переменную длину команды.

Как альтернативу, рассмотрим микроконтроллеры компании Microchip Technology Inc, которая была одной из первых массовых производителей этих изделий. Популярное среднее семейство микроконтроллеров PIC16 имеют простую и эффективную систему команд, состоящую всего из 35 команд.

Каждая команда МК состоит из одного 14 - разрядного слова, разделенного на код операции (OPCODE), определяющий тип команды и одного или нескольких операндов, определяющего операцию команды /5/.

Система команд аккумуляторного типа, ортогональна и разделена на три основных группы:

- байт ориентированные команды;
- бит ориентированные команды;
- команды управления и операций с константами.

Для байт-ориентированных команд 'f' является указателем регистра, а 'd' указателем адресата результата. Указатель регистра определяет, какой регистр должен использоваться в команде. Указатель адресата определяет, где будет сохранен результат. Если 'd'=0, результат сохраняется в регистре W. Если 'd'=1, результат сохраняется в регистре, который используется в команде.

В бит-ориентированных командах 'b' определяет номер бита участвующего в операции, а 'f' - указатель регистра, который содержит этот бит.

В командах управления или операциях с константами 'k' представляет восемь или одиннадцать бит константы или значения литералов.

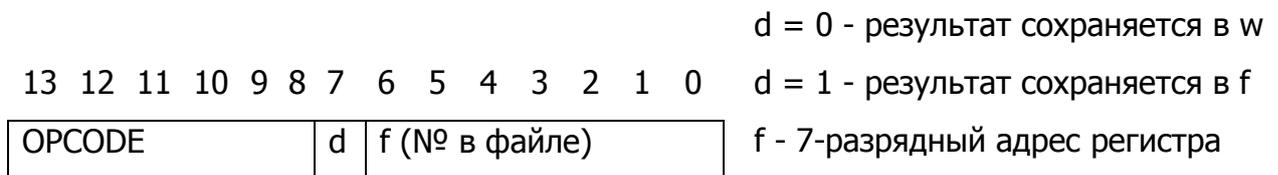
Все команды выполняются за один машинный цикл, кроме команд условия, в которых получен истинный результат и инструкций изменяющих значение счетчика команд PC. В случае выполнения команды за два машинных цикла, во втором

Микропроцессорная техника

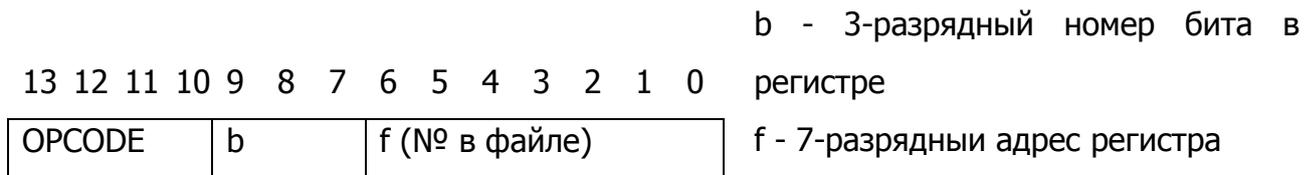
цикле выполняется инструкция – нет операций (NOP). Один машинный цикл состоит из четырех тактов генератора. Для тактового генератора с частотой 4 МГц все команды выполняются за 1 мкс, если условие истинно или изменяется счетчик команд PC, команда выполняется за 2 мкс.

На рис. 5.3 показан формат трех групп команд. Код операции может быть от 3 до 6 бит, что позволяет реализовать 35 команд. Любой не реализованный код операции сохранен для последующих разработок. Использование недокументированного кода операции может привести к непредсказуемым результатам.

Байт ориентированные операции с регистрами

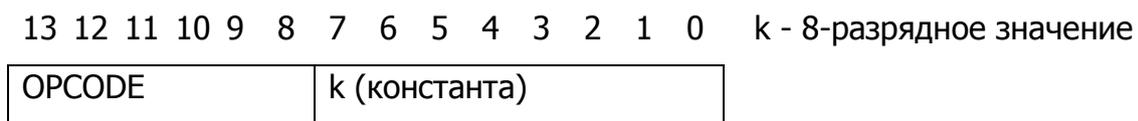


Бит ориентированные операции с регистрами



Команды управления и операций с константами

Общее



Только для инструкций CALL и GOTO

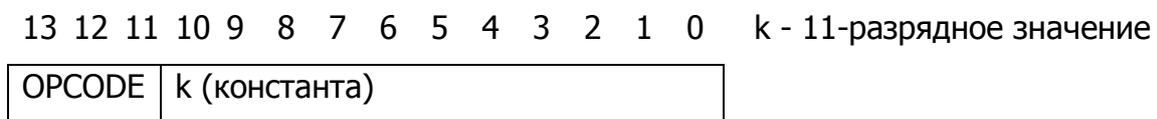


Рис. 5.3 – Общий формат команд микроконтроллеров среднего семейства

Контроллеры имеют постоянный размер команды (14 бит) и переменный код операции 3, 4 или 6 бит.

Рассмотренные процессор i8080 и контроллеры семейства PIC16 имеют восьмиразрядную память данных и сопоставимую тактовую частоту.

## **6. Перспективы применения микропроцессорной техники в системах управления**

Особенность современного этапа развития техники и технологий заключается в том, что в современном машиностроении преобладает интеграция электрических, электронных и механических компонентов. Это влечёт за собой и изменения к требованиям, предъявляемым к проектированию, изготовлению и эксплуатации техники. На первое место выходят навыки и знания в обращении с электронной техникой управления с использованием компьютеров и микроконтроллеров; требуется владение электропневмоавтоматикой, электрогидроавтоматикой и электромеханикой во всех их компонентах. Предприятия машиностроения всё в большей степени нуждаются в специалистах – универсалах, понимающих как компьютерные системы управления, так и электромеханику, гидropневмоавтоматику.

Решению задач подготовки таких специалистов предназначена дисциплина «Микропроцессорная техника». Это обучение в области науки и техники, посвящённое созданию и эксплуатации машин и систем с компьютерным управлением движением. Собственно говоря, любая современная машина, станок, самолёт или автоматическая линия — это воплощение мехатроники. Мехатроника базируется на знаниях в области механики, электроники, информатики и микропроцессорной техники. Акцент должен быть сделан на взаимодействии механики, электроники и микропроцессорной техники между собой.

Современные технологии для систем управления движением объединяют новые аппаратные решения, ставшие возможными благодаря достижениям микроэлектроники, и соответствующее программное обеспечение, построенное на основе информационных технологий. Электронные модули должны выполнять заданные управляющие, преобразующие и информационные функции и при этом конструктивно быть встроенными в мехатронные модули. Развитие встраиваемых систем управления технологическим оборудованием и процессами невозможно без применения микроконтроллеров. Эта разновидность микропроцессорных систем за последнее время проходит этап бурного технологического развития. Современные микроконтроллеры по разрядности и производительности догнали, а в мобильных

### Микропроцессорная техника

приложения и обогнали современные универсальные процессоры при существенно меньшем энергопотреблении и габаритах.

Кроме стандартных частей микропроцессорная система содержит и некоторые индивидуальные схмотехнические части для сопряжения модулей и управления ими. Такие индивидуальные части системы ранее приходилось строить с помощью ИС малого и среднего уровней интеграции, применение которых ведет к резкому увеличению числа корпусов и внешнего монтажа в схемах МПС, а это сопровождается столь же резким снижением их быстродействия и надежности, росту габаритных размеров и потребляемой мощности.

Таким образом, для создания высококачественной сложной цифровой аппаратуры следует использовать интегральные схемы высокого уровня интеграции.

Увеличивается использование БИС/СБИС в специализированных частях микропроцессорных систем по мере развития программируемых логических интегральных схем (ПЛИС). Применение ПЛИС с программируемой логикой ускоряет проектирование изделий, уменьшает вероятность ошибок, габариты и энергопотребление.

Совершенствуется прикладное программное обеспечение микро- ЭВМ и контроллеров. Многие ведущие производители комплектующих, для разработки систем управления, предлагают программное обеспечение и специальные отладочные платы. Популярным стало использование адаптированного для разработки ПО систем управления языка C++.

Массовость производства микропроцессорных систем управления и их невысокая стоимость постепенно вытесняет классические аналоговые системы заменяя их цифровыми практически во всех отраслях техники.

## Литература

1. Новиков Ю.В., Скоробогатов П. К. Основы микропроцессорной техники. Учебное пособие. — 4-е изд. М.: ИУИТ; БИНОМ, 2011. — 358 с.
2. Максимов Н.В., Партыка Т.Л., Попов И.И. Архитектура ЭВМ и вычислительные системы. Учебник. — 5-е изд., перераб. и доп. — М.: Форум, Инфра-М, 2013. — 512 с.: ил.
3. Гусев В.Г., Гусев Ю.М. Электроника и микропроцессорная техника. Учебник — 6-е изд., стер. — М.: КНОРУС, 2013. — 800 с. — (Бакалавриат).
4. Рассадкин Ю.И., Сеницын А.В., Бошляков А.А. (глава 2.4.3). Микропроцессорные системы управления в робототехнике и мехатронике. Учебный курс. Московский государственный технический университет им. Н.Э.Баумана, кафедра СМ-7 "Специальная робототехника и мехатроника".  
<http://bigor.bmstu.ru/?cnt/?doc=MPSU/base.cou>
5. Microchip. Главная страница. <http://www.microchip.com/ru/>
6. Atmel. Главная страница. <http://www.atmel.com/ru/ru/>
7. ГОСТ 26765.51-86. Интерфейс магистральный параллельный МПИ системы электронных модулей. Общие требования к совокупности правил обмена информацией.
8. А.В. Соловьев, Н.Ю. Ершова. Организация вычислительных систем. Электронная книга (fb2 – 945,6 Кб, txt – 329,5 Кб, html - 944 Кб, epub – 1,1 Мб). Лекция 14: Интерфейсы вычислительных систем.  
<http://www.intuit.ru/studies/courses/92/92/lecture/2801?page=2>.