

Федеральное агентство по образованию
Государственное образовательное учреждение
высшего профессионального образования
Владимирский государственный университет

О.В. ВЕСЕЛОВ

ЛОКАЛЬНЫЕ СЕТИ МАЛЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

Учебное пособие

«В печать»:

Автор –

О.В. Веселов

Зав. кафедрой –

И.Н. Егоров

Редактор –

И.А. Арефьева

Начальник РИО –

Е.П. Викулова.

Директор издательства –

Ю.К. Жулев.

Владимир 2005

УДК 681.327
ББК 32.988-5
В38

Рецензенты:

Кандидат технических наук, доцент зав. кафедрой
электротехники, электроники и автоматики Московского
государственного технологического университета «Станкин»
В.В. Филатов

Доктор технических наук, профессор
директор научно-производственного предприятия «Энергоприбор»
В.К. Новиков

Печатается по решению редакционно-издательского совета
Владимирского государственного университета

Веселов, О.В.

В38 Локальные сети малых вычислительных систем : учеб. пособие /
О. В. Веселов ; Владим. гос. ун-т. – Владимир : Изд-во Владим. гос.
ун-та, 2005. – 212 с. – ISBN 5-89368-631-4.

Разработано в соответствии с Государственным образовательным стандартом Министерства образования Российской Федерации по специальностям 220301 (210200) – автоматизация технологических процессов и производств и 220401 (071800) – мехатроника. Рассмотрены методы и средства построения сетей малых вычислительных систем. Приведены модель открытых систем как единая основа построения локальной сети, технические средства реализации систем связи, устройств и протоколов обмена информацией.

Предназначено для студентов специальностей 220401 (071800) и 220301 (210200) дневной формы обучения. Может быть полезно студентам специальностей 220201 (210100), 151001 (120100), 200503 (072000), 220402 (210300) и др.

Табл. 38. Ил. 91. Библиогр.: 14 назв.

УДК 681.327
ББК 32.988-5

ISBN 5-89368-631-4

© Владимирский государственный
университет, 2005

ОГЛАВЛЕНИЕ

Введение.....	5
Глава 1. МАЛЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И ОСНОВЫ ЛОКАЛЬНЫХ СЕТЕЙ.....	9
1.1. Общая характеристика малых вычислительных систем.....	9
1.2. Понятие и характеристика интерфейсов.....	11
1.3. Основные характеристики ЛВС.....	14
1.4. Базовая модель взаимодействия систем.....	20
Контрольные вопросы.....	32
Глава 2. КОНФИГУРАЦИЯ ЛВС.....	33
2.1. Шинная топология ЛВС.....	33
2.2. Кольцевая ЛВС.....	34
2.3. Звездообразная конфигурация ЛВС.....	36
2.4. Древоподобные ЛВС.....	38
2.5. Петлевые и полносвязные конфигурации.....	39
2.6. Качественные характеристики ЛВС.....	40
Контрольные вопросы.....	43
Глава 3. ФИЗИЧЕСКАЯ РЕАЛИЗАЦИЯ СРЕДЫ ПЕРЕДАЧИ ДАННЫХ.....	44
3.1. Общая характеристика каналов связи.....	44
3.2. Витая пара.....	47
3.3. Оптический кабель.....	62
3.4. Коаксиальный кабель.....	81
Контрольные вопросы.....	83
Глава 4. ПОВТОРИТЕЛИ, КОММУТАТОРЫ.....	84
4.1. Техническая реализация и дополнительные функции коммутаторов.....	84
4.2. Коммутаторы на основе коммутационной матрицы.....	85
4.3. Коммутаторы с общей шиной.....	86
4.4. Коммутаторы с разделяемой памятью.....	88
4.5. Комбинированные коммутаторы.....	89
4.6. Конструктивное исполнение коммутаторов.....	89
4.7. Репитеры.....	90
Контрольные вопросы.....	95
Глава 5. ПОСЛЕДОВАТЕЛЬНЫЕ ИНТЕРФЕЙСЫ RS-232 и RS-485.....	102
5.1. Интерфейс RS-232C.....	102
5.2. Интерфейс RS-485.....	113

5.3. Согласование и конфигурация линии связи.....	116
5.4. Защитное смещение.....	118
5.5. Исключение приема при передаче в полудуплексном режиме....	120
5.6. "Горячее" подключение к линии связи.....	121
5.7. Рекомендации по организации протокола связи.....	122
5.8. Программные методы борьбы со сбоями.....	125
5.9. Защита устройств от перенапряжений в линии связи.....	129
5.10. Дополнительные меры защиты от помех.....	131
Контрольные вопросы.....	139
Глава 6. ПРОТОКОЛЫ CAN И PROFIBUS.....	140
6.1. CAN-протокол.....	140
6.2. Основы PROFIBUS (PROcess FieLd BUS).....	169
Контрольные вопросы.....	191
Глава 7. ЛОГИЧЕСКАЯ ОРГАНИЗАЦИЯ СЕТИ	193
7.1. Протокол работы сети.....	198
7.2. Управление дискретными сигналами.	200
7.3. Управление двигателями постоянного тока, чтение датчиков положения.....	203
7.4. Программа управления протоколом.....	207
Контрольные вопросы.....	209
Заключение.....	210
Библиографический список.....	211

ВВЕДЕНИЕ

По мере совершенствования компьютеров, увеличения их числа, развития и усложнения средств программного обеспечения, в том числе и прикладного, увеличения объемов данных возникла насущная необходимость в соединении вычислительных устройств между собой. Такое связывание позволяет объединить их ресурсы – процессоры, память, каналы связи – и представляет собой вычислительную сеть, в которой каждое вычислительное устройство может передать другому, подключенному к сети, любой набор данных.

Таким образом, вычислительная сеть, объединяющая ресурсы нескольких вычислительных устройств, позволяет каждому из них использовать всю совокупность этих ресурсов. В такой сети отдельные пользователи получают возможность не только обмениваться файлами и сообщениями, использовать общие периферийные устройства, но и, что важно, использовать информацию, рассредоточенную в сети по отдельным вычислительным устройствам, при решении своих задач. Этот подход к совместному применению ресурсов в сети составляет основу технологии открытых сетевых вычислений ONC (Open Network Computing).

В зависимости от того, на каком расстоянии друг от друга находятся вычислительные устройства, объединенные в сеть, различают локальные, региональные и глобальные вычислительные сети. Если локальная вычислительная сеть (ЛВС) объединяет вычислительные устройства, находящиеся в одном или соседних помещениях, в пределах одного или соседних зданий на расстояниях, не превышающих одного-двух километров, региональная – связывает абонентов, расположенных на значительном расстоянии друг от друга в пределах города, района или области, то глобальная сеть – это сеть, в которой объединены компьютеры, расположенные в различных странах или на различных континентах земного шара (примером может служить Интернет).

В свою очередь, информационно-вычислительные сети могут быть подразделены на ряд типов в зависимости от их назначения.

В литературе достаточно широко описаны компьютерные сети, и большая часть информации уделяется глобальным сетям. Локальные сети если и описываются, то с позиции корпоративных сетей, предназначенных для решения информационно-экономических задач и использующих аппаратуру такую же, как и в глобальных сетях. Приводимые сведения охватывают теоретические вопросы построения систем, оставляя без внимания прикладные аспекты. Однако сети для реализации управления процессами и производствами на основе малых вычислительных систем (МВС) описаны недостаточно, что делает актуальным написание этой книги.

Стремительное развитие микропроцессорной техники предопределило множество направлений в организации взаимодействия МВС с использованием разнородных специализированных систем связи. С одной стороны, желание повысить производительность МВС породило развитие таких устройств, как USB, CAN-bus, IEE1394, составивших альтернативу уже известным системам связи, таким как Multibus, Profitbus и т.п., с другой – позволило найти решение, при котором реализация систем коммуникаций становится простой и доступной.

Существует специфика построения систем управления, согласно которой управляющая ЭВМ должна быть связана с МВС, выполняющей задачи управления на исполнительном уровне. Учитывая структуру микроконтроллеров, входящих в состав МВС, обеспечить связь между ЭВМ и МВС, в которых нет специальных устройств, подобных сетевым платам, возможно лишь с использованием широко известного коммуникационного порта RS-232 или же шин вышеперечисленных. Еще одно обстоятельство – возможность подключения большого числа МВС к единой магистрали без использования дополнительных устройств. Это связано с тем, что необходимо иметь возможность строить децентрализованные системы управления, в которых связывают управляющую ЭВМ с пространственно разнесенными МВС. Если ранее использовалась шина с параллельной передачей информации в силу ее высокой производительности, то последовательный канал сегодня имеет не меньшую производительность и обладает существенным преимуществом – минимальным числом проводников (как правило, два проводника) для реализации системы связей. А в условиях пространственно распределенных систем это единственно правильное решение.

Вычислительные сети, в зависимости от способов взаимодействия вычислительных устройств в них, можно разделить на две группы – централизованные и одноранговые.

Централизованные сети строятся на основе архитектуры "клиент – сервер", которая предполагает выделение в сети так называемых "серверов" и "клиентов". К клиентам относятся рабочие станции сети, которые не имеют непосредственных контактов друг с другом и могут общаться между собой только через сервер, следовательно, одна станция не может использовать файл, находящийся на другой станции. Для этого файл должен находиться на файл-сервере. Все локальные ресурсы рабочих станций так и остаются локальными и, следовательно, не могут быть использованы никакой другой рабочей станцией.

Серверы управляют подключенными к ним общими разделяемыми ресурсами сети. В качестве сервера может быть использован или обычный персональный компьютер, или же это может быть специализированное устройство, реализующее одну из указанных функций управления. В этом последнем случае специализированный сервер по своему внешнему виду может отличаться от обычного персонального компьютера.

С точки зрения пользователей архитектура "клиент – сервер" предоставляет им быстрый и простой доступ с локальной рабочей станции к информации и функциям, содержащимся где-то в сети. Среди других достоинств, важных для пользователя, можно отметить повышение производительности, возможность свободного выбора необходимого программного обеспечения, простоту в использовании инструментальных средств.

Одноранговые ЛВС основаны на равноправной (peer-to-peer) модели взаимодействия компьютеров. В такой сети каждый компьютер может выступать, смотря по обстоятельствам, и как рабочая станция, и как сервер, предоставляющий возможность доступа другим коллегам к своим ресурсам. В одноранговой сети рабочие станции могут напрямую связываться друг с другом (в отличие от централизованной структуры сети) и совместно использовать те из них, которые на данном конкретном компьютере определены как разделяемые. Сети подобного типа недороги, поскольку не требуют приобретения дорогостоящего выделенного сервера, просты в эксплуатации и могут быть рекомендованы для малочисленных групп пользователей, которым не требуются большие вычислительные ресурсы.

Для обеспечения межсетевого взаимодействия служат специальные устройства: мосты, маршрутизаторы, коммутаторы и шлюзы.

Для организации работы вычислительной сети служат сетевые операционные системы (ОС). При выборе сетевой ОС принимают во внимание несколько факторов: требования к производительности сети, надежности и степени ее информационной безопасности, требуемые ресурсы памяти; функциональная мощность; простота эксплуатации, возможность объединения с другими сетями; цена.

Для создания одноранговых сетей используются следующие пакеты программных средств, ориентированные на простые и недорогие решения: Coactive Connector, Desk to Desk, LAN Link, LANtastic и др. При выборе решения для создания одноранговой сети нельзя ограничиваться рассмотрением только программного аспекта. Необходимо также решать проблему выбора средств аппаратной поддержки: некоторые сетевые программные средства позволяют использовать действующий промышленный стандарт, обычно Ethernet, в то время как другие ориентированы на использование собственных аппаратных средств.

Какая сеть лучше – одноранговая или построенная по типу "клиент – сервер"? Ответ на этот вопрос зависит от потребностей пользователей, финансовых возможностей и опыта разработчиков.

Глава 1. МАЛЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И ОСНОВЫ ЛОКАЛЬНЫХ СЕТЕЙ

1.1. Общая характеристика малых вычислительных систем

Невозможно описать вычислительную систему в нескольких словах. Такие системы весьма сложны, и только многолетний опыт работы с ними позволяет обрести глубокое понимание их структуры. Вычислительные системы различаются своими размерами, производительностью, конфигурацией и степенью специализации. Область их применения постоянно растет.

Так что же такое вычислительная система? Отметим лишь несколько примеров ее описания:

1. Это совокупность электронных узлов, используемых для обработки информации.

2. Обрабатываемая информация поступает в систему от устройств ввода, таких как термометры, датчики времени и давления, вольтметры, счетчики, весы, телефонные линии и т. п.

3. После обработки введенной информации результаты передаются на одно или несколько устройств вывода, которые также весьма разнообразны. Среди них – клапаны в трубопроводах, телефонные линии, электродвигатели, индикаторы, позволяющие контролировать состояние системы или объекта, которым она управляет, сигналы управления транспортным потоком и т. п.

4. При большом объеме перерабатываемой информации вычислительные системы объединяются в сеть, обеспечивая тем самым распараллеливание задач.

Таким образом, микроЭВМ состоит из электронного оборудования для обработки информации, нескольких устройств – источников информации (устройств ввода) и нескольких устройств – приемников информации (устройств вывода). Соединение этих устройств в вычислительной системе проиллюстрировано на рис. 1.1.

Устройства ввода и вывода подсоединены к устройству обработки информации через канал связи, называемый *шиной ввода-вывода*, по которой происходит обмен данными между устройствами. Принято говорить об устройствах ввода-вывода как о *внешних* устройствах.

Устройство обработки информации состоит из двух основных подсистем: собственно обрабатывающего устройства (*центрального процес-*

сорного устройства, или просто процессора) и устройства хранения информации – устройства памяти.

Память необходима потому, что процессор ведет обработку информации гораздо быстрее, чем могут передавать ее внешние устройства. Информация, необходимая для решения задачи, вводится в память с внешних устройств. Затем процессор обрабатывает эту информацию и помещает результаты в память. Далее результаты передаются на устройство вывода (скорость вывода относительно мала). Память построена на быстродействующих электронных компонентах, что позволяет согласовать ее по скорости работы с процессором.

Для объединения двух и более вычислительных устройств используются различные системы физической связи и их различная логическая организация.

На рис. 1.1 показаны два варианта организации связи между памятью и остальными компонентами системы. Оба этих варианта применяются в нескольких серийных моделях малых и микроЭВМ. Различие здесь состоит в том, что в одной из систем (см. рис. 1.1, верхняя часть рисунка) используется специальная шина связи между процессором и памятью, а в другой – память находится на той же шине, что и внешние устройства. Последняя из этих схем применяется в большинстве микропроцессорных систем, а также в некоторых новейших моделях малых ЭВМ.

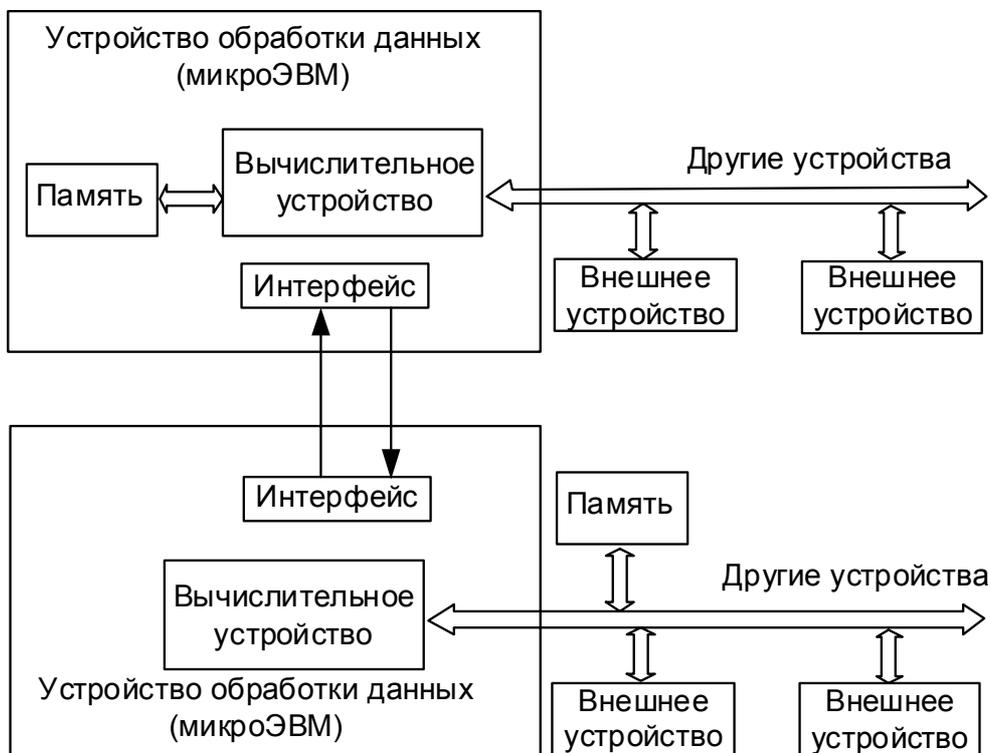


Рис. 1.1. Блок-схема вычислительной системы

Универсальность цифровых вычислительных систем объясняется двумя причинами. Во-первых, к такой системе могут быть подключены самые разнообразные внешние устройства. Во-вторых, что гораздо более существенно, центральный процессор может быть запрограммирован для выполнения требуемой задачи не в момент изготовления, а в процессе использования системы.

Производительность и гибкость вычислительной системы определяются именно тем обстоятельством, что программы хранятся в памяти. В зависимости от внешних событий система может переходить от одной программы к другой. Иными словами, поведение системы может изменяться в зависимости от изменения состояния процесса.

Таким образом, конструкция вычислительной системы позволяет настраивать ее на решение определенных задач:

- путем подключения к ней требуемых внешних устройств;
- объединения нескольких устройств в вычислительную среду;
- написания *программистом* последовательности кодов – программы, которая хранится в памяти ЭВМ.

1.2. Понятие и характеристика интерфейсов

В отечественной практике для описания совокупности схемотехнических средств и функций, обеспечивающих непосредственное взаимодействие составных элементов систем управления и обработки данных, сетей, систем передачи данных, подсистем периферийного оборудования, используются понятия «интерфейс», «стык», «протокол» [1].

Под *стандартным интерфейсом* понимается совокупность унифицированных аппаратурных, программных и конструктивных средств, необходимых для реализации взаимодействия различных функциональных элементов в автоматических системах при условиях, предписанных стандартом и направленных на обеспечение информационной, электрической и конструктивной совместимости указанных элементов.

Понятие *стык* согласно ГОСТ 23633-79 означает «место соединения устройств передачи сигналов данных, входящих в системы передачи данных». Оно используется вместо понятия «интерфейс» для описания функций и средств сопряжения элементов средств связи и СПД.

Под понятием *протокол* понимается строго заданная процедура, или совокупность правил, регламентирующая способ выполнения определенного класса функций.

Взаимосвязь понятий «интерфейс» и «протокол» не всегда однозначна. С одной стороны, практически любой интерфейс содержит в большей или меньшей степени элементы протокола, определяемые процедурными и функциональными характеристиками интерфейса. В этом отношении понятие «интерфейс» шире понятия «протокол». С другой стороны, существуют протоколы, например протоколы управления обменом данными и по каналам связи, которые представляют собой достаточно самостоятельный компонент звена, системы или сети. Многие функции таких протоколов не вписываются в функции интерфейса. Кроме того, сфера действия таких протоколов в системе или сети охватывает несколько различных интерфейсов (стыков). В подобных случаях понятие «протокол» оказывается шире понятия «интерфейс».

Близость понятий «интерфейс» и «протокол» подчеркивается тем, что во многих случаях одна и та же совокупность средств и функций взаимодействия называется в разных источниках по-разному: интерфейсом или протоколом (например, рекомендация X.25 МККТТ). В силу этого и сложившихся традиций средства и функции взаимодействия некоторых локальных систем или сетей (например, КАМАК, ИЛПС, PROWAY) определяются понятием «интерфейс», для других подобных систем или сетей (например, стандартизируемых в МОС ПМС 8802) – понятием «протокол».

Комплексность понятий «интерфейс» и «протокол» характеризуют также используемые термины *интерфейсная система, система протоколов, физический интерфейс, логический интерфейс* и т. д.

Основное назначение интерфейсов, стыков и протоколов — унификация внутри- и межсистемных, внутри- и межсетевых связей с целью эффективной реализации методов проектирования функциональных элементов (ФЭ) вычислительных систем (ВС), СОД и сетей. Основные функции интерфейсов и стыков – обеспечение информационной, электрической и конструктивной совместимости между ФЭ системы или сети.

Информационная совместимость – согласованность взаимодействия ФЭ в соответствии с совокупностью логических условий. Логические условия определяют: 1) структуру и состав унифицированного набора шин; набор процедур по реализации взаимодействия и последовательность их выполнения для различных режимов функционирования; способ кодирования и форматы команд, данных, адресной информации и информации

состояния; временные соотношения между управляющими сигналами, ограничения на их форму и взаимодействие; 2) в целом функциональную и структурную организацию интерфейса. Условия информационной совместимости влияют на объем и сложность схемотехнического оборудования и программного обеспечения, а также на основные технико-экономические показатели интерфейса.

Электрическая совместимость — согласованность статических и динамических параметров электрических сигналов в системе шин с учетом ограничений на пространственное размещение устройств интерфейса и техническую реализацию приемопередающих элементов (ППЭ).

Условия электрической совместимости определяют: тип ППЭ; соотношение между логическими и электрическими состояниями сигналов и пределы их изменения; коэффициенты нагрузочной способности ППЭ и значения допустимой емкостной и резистивной нагрузки в устройстве; схему согласования линии; допустимую длину линии и порядок подключения линии к разъемам; требования к источникам и цепям электрического питания; требования по помехоустойчивости.

Большинство условий электрической совместимости и типы ППЭ обычно регламентируются стандартом. Условия электрической совместимости влияют на основные показатели интерфейса, в частности на скорость обмена данными, предельно допустимое число подключаемых устройств, их конфигурацию и расстояние между устройствами, помехозащищенность.

Конструктивная совместимость — согласованность конструктивных элементов интерфейса, предназначенных для обеспечения механического контакта электрических соединений и механической замены схемных элементов, блоков и устройств.

Условия конструктивной совместимости определяют типы: соединительных элементов (разъем, штекер и распределение линий связи внутри соединительного элемента); конструкции платы, каркаса, стойки; конструкции кабельного соединения.

Условия конструктивной совместимости в рекомендациях стандартных интерфейсов не всегда установлены полностью, а только в некоторых сферах деятельности; стандартизация физических системных параметров совместно с необходимыми протоколами взаимосвязи и взаимодействия вычислительных машин и подключаемых технических средств (за исключением средств телеобработки и сетей передачи данных) с использованием различных физических средств связи.

1.3. Основные характеристики ЛВС

Основные понятия

В общепринятом понимании локальная вычислительная сеть (Local area network – LAN) – система связей, использующая средства и методы передачи данных, расположенная на небольшой территории (здание, предприятие, учреждение) и ориентированная на эту территорию.

Шинная сеть (Bus network) – это ЛВС, в которой имеется только один маршрут между любыми двумя станциями, и данные, передаваемые любой станцией, одинаково доступны всем станциям сети. Такая сеть может иметь линейную, радиальную или древовидную конфигурацию. В двух последних случаях в каждом оконечном узле размещены станции данных, а в промежуточных узлах – повторители, соединители, усилители и разветвители.

Кольцевая сеть (Ring network) – ЛВС, обеспечивающая однонаправленную передачу данных между станциями по одной физической среде с возвратом данных к передающей станции.

Звездно-кольцевая сеть (Star/ring network) – кольцевая сеть с однонаправленной передачей, в которой несколько сгруппированных станций подключены к физической среде посредством интерфейсных модулей.

Локальная вычислительная сеть основной полосы частот (Baseband LAN) – ЛВС, в которой данные кодируются и передаются без модуляции несущей.

В *широкополосной ЛВС* (Broadband LAN) используются методы разделения каналов и модуляции.

Физическая среда (Transmission medium) — физический материал, по которому с высокой скоростью перемещаются данные между подключаемыми станциями ЛВС (например, симметричный кабель, коаксиальный кабель, волоконно-оптический кабель).

Физический уровень (Physical layer) — уровень, ответственный за сопряжение станции с физической средой, распознавание и преобразование сигналов, выдаваемых физической средой, и данных, принимаемых из подуровня УДС.

Уровень звена данных – *УЗД* (Data link layer – DLL) – концептуальный уровень в иерархической архитектуре оконечной системы (станции данных), обеспечивающий протокол управления звеном данных, формирование и передачу кадров данных.

Управление доступом к среде – УДС (Medium access control – MAC) – нижний подуровень уровня звена данных, управляющий доступом станций к физической среде с учетом особенностей физической среды.

Управление логическим звеном – УЛЗ (Logical link control – LLC) – верхний подуровень уровня звена данных, выполняющий не зависящие от физической среды и метода доступа к ней функции звена данных.

Диспетчер станции – ДИСП (Station management) – концептуальный элемент управления станции, взаимодействующий со всеми уровнями станции и ответственный за установку и сброс управляющих параметров, получение отчетов об ошибочных ситуациях и определение необходимости подключения станций к сети и отключения их от сети.

Передача физических сигналов – ПФС (Physical signaling – PLS) – верхний подуровень физического уровня, выполняющий бит – символическое кодирование и декодирование, прием и передачу данных и некоторые функции изоляции УДС от особенностей физической среды.

Интерфейс с устройством сопряжения – ИУС (Attachment unit interface – AUI) – интерфейс между модулем сопряжения со средой и оконечным оборудованием данных (ООД) внутри станции.

Модуль сопряжения со средой – МСС (Medium attachment unit – MAU) – устройство, используемое в станции для сопряжения ООД с физической средой.

Модуль доступа к среде – МДС (Physical medium attachment – PMA) – часть МСС, содержащая функциональные схемы и используемая для подключения станции к физической среде и отключения станции от физической среды без нарушения работы станции.

Интерфейс, зависящий от среды, – ИЗС (Medium dependent interface – MDI) – механический и электрический интерфейс между физической средой и МСС.

Назначение и особенности ЛВС

Вначале ЛВС применялись главным образом в университетах и научно-исследовательских институтах. В течение сравнительно короткого периода времени производство и применение ЛВС получило широкое распространение, чему способствовали следующие факторы: развитие микропроцессорной техники, гибких магнитных дисков, снижение цен на электронные компоненты, расширение возможностей терминальных устройств, рост объема передаваемой и обрабатываемой информации и, как следствие, увеличение числа средств в учреждениях, на предприятиях, заводах и

т.п. Важную роль сыграл также тот факт, что в условиях действия перечисленных факторов большая часть вырабатываемой и потребляемой информации замыкалась внутри предприятия.

По своим размерам и конфигурациям ЛВС занимают промежуточное положение между обычными вычислительными сетями и подсистемами ввода-вывода ЭВМ, размещенными в одном помещении или разнесенными на сотни метров. В то же время ряд особенностей позволяет выделить ЛВС в отдельный класс вычислительных сетей. К этим особенностям относятся: размещение ЛВС на сравнительно небольшой территории; простые методы модуляции сигналов, возможность передачи немодулированных сигналов, низкий уровень ошибок и простые интерфейсные устройства вследствие малых расстояний; отсутствие ограничений, налагаемых сетями общего пользования; соединение в ЛВС самых разнообразных и независимых устройств; простота изменения конфигурации и самой среды передачи; низкая стоимость сети передачи данных по сравнению со стоимостью подключаемых устройств.

Расстояния, охватываемые ЛВС, могут существенно различаться: от сотен метров до десятков километров. Главное же отличие локальных сетей от глобальных — наличие единого для всех абонентов высокоскоростного канала передачи данных.

Термин "высокоскоростной канал" имеет условный характер, поскольку скорость передачи данных целесообразно оценивать по отношению к подключенным устройствам. Скорости передачи в ЛВС должны быть существенно выше скоростей, требуемых для отдельных устройств. Примеры соотношений указанных скоростей приведены в таблице.

Скорости передачи данных в ЛВС

Устройство	Скорость, кбит/с	
	устройства	ЛВС
Большая ЭВМ	$20 \cdot 10^3$	$50 \cdot 10^3$
Мини- и терминальная ЭВМ	$1 \cdot 10^3$	$5 \cdot 10^3$
МикроЭВМ	300	500
Терминал	20	50
Калькулятор	1	5

Важная особенность высокоскоростных систем – неравномерное распределение нагрузки. Отношение пиковой нагрузки к средней может со-

ставлять 1000 и даже выше. Для таких систем эффективен метод коммутации пакетов, который принят в качестве основного для большинства современных ЛВС. Через ЛВС пакет данных проходит за несколько миллисекунд, время же его передачи через глобальную сеть составляет около секунды.

Локальные вычислительные сети обеспечивают более дешевый способ комплектации оборудования. Низкая вероятность возникновения ошибок позволяет упростить сетевые протоколы, а высокая скорость передачи повышает эффективность коллективного доступа многих пользователей к общему банку данных.

Несмотря на то, что основной областью применения ЛВС является передача данных, многие используемые в ЛВС методы могут быть распространены также на передачу речевой, текстовой и видеоинформации, что, например, позволяет объединить многие формы учрежденческой связи в рамках одной сети.

Некоторые последние разработки ЛВС базируются на системах с передачей модулированных сигналов и разделением имеющейся полосы частот на отдельные высокоскоростные каналы. Такие сети известны как широкополосные, и во многих случаях каждую такую сеть можно рассматривать как несколько отдельных независимых ЛВС, объединенных общим кабелем.

В общем случае ЛВС представляет собой корпоративную коммуникационную систему, позволяющую однотипным или разнородным средствам вычислительной техники сообщаться друг с другом с помощью единой передающей среды. Связь может осуществляться между большими, малыми и микроЭВМ, специализированными процессорами, персональными ЭВМ, терминалами и терминальными станциями, а также специализированными средствами (устройства связи с объектом и т.п.). При этом ЛВС обеспечивает простое и удобное объединение всех средств в пределах решаемой задачи.

К наиболее важным характеристикам ЛВС, отличающим их друг от друга, относятся конфигурация, или топология сети, тип передающей физической среды, скорость передачи сигналов, метод доступа станций ЛВС к среде, архитектура сети.

В настоящее время ЛВС стали одним из самых распространенных средств передачи и обработки данных, поскольку по своим характеристикам и возможностям они наиболее полно отвечают потребностям предприятий, занимающихся управлением, планированием и производством.

Требования к ЛВС

Основопологающие требования к ЛВС, ориентированным на передачу данных, сформулированы в 1981 г. комитетом IEEE (Institute of Electrical and Electronic Engineers) и опубликованы в виде проекта стандарта [1]. Последующая практика разработки и применения ЛВС в основном подтвердила правомерность этих требований, хотя и внесла в них коррективы, отраженные в [2, 3] и других работах.

Приводимый ниже перечень требований к ЛВС основан на анализе упомянутых источников. Не все из перечисленных ниже требований являются общими для всех типов ЛВС. Степень выполнимости некоторых из них существенно зависит от назначения, способов использования ЛВС и ряда других факторов.

Общие требования:

- выполнение разнообразных функций по передаче данных, включая пересылку файлов, поддержку терминалов (в том числе высокоскоростных графических), электронную почту, обмен с внешними ЗУ, обработку сообщений, доступ к файлам и базам данных;
- возможность подключения большого набора стандартных и специальных устройств, в том числе больших, малых и персональных ЭВМ, терминалов, оборудования контроля и управления и др.;
- возможность подключения как современных и перспективных, так и ранее разработанных устройств с различными программными средствами, архитектурой, принципами работы;
- доставка пакетов адресату с высокой достоверностью при обеспечении виртуальных соединений и датаграммной службы;
- обеспечение непосредственной связи между подключенными устройствами без промежуточного накопления и хранения данных;
- простота монтажа, модификации и расширения сети;
- возможность поддержки большого числа устройств с помощью одной ЛВС и охвата территории диаметром не менее 2 км;
- соответствие по возможности существующим стандартам.

Требования к взаимодействию устройств в сети:

- возможность для каждого устройства связываться и взаимодействовать с любым другим устройством;
- обеспечение равноправного доступа к физической среде для всех коллективно использующих ее устройств;

- возможность адресации пакетов одному устройству, группе устройств, всем подключенным устройствам;
- обеспечение возможности некоторым пользователям назначать и менять собственный адрес (в рамках ограничений, сохраняющих целостность сети).

Информационные требования:

- обеспечение "прозрачного" режима обслуживания, возможность приема, передачи и обработки любых сочетаний бит, слов и символов, в том числе не кратных восьми (последнее требование противоречит позиции IEEE, согласно которой блок данных должен содержать целое число октетов, однако для некоторых применений это требование IEEE неприемлемо);
- отсутствие существенного снижения пропускной способности сети при достижении ее полной загрузки, и даже перегрузки, во избежание длительной блокировки сети;
- небольшая по величине фиксированная задержка передачи пакета через ЛВС.

Требования к надежности и верности:

- отказ или отключение питания подключенного устройства вызывают только переходную ошибку;
- сеть не должна находиться в состоянии неработоспособности более 0,02 % от полного времени работы (это составляет около 20 мин простоя в год для учрежденческой системы и около 2 ч для непрерывно функционирующей системы);
- средства обнаружения ошибок выявляют все пакеты, содержащие до четырех искаженных бит. Если же верность передачи достаточно высока, сеть сама не исправляет обнаруженные ошибки; функции анализа, принятия решения и исправления ошибок выполняются подключенными устройствами;
- сеть обнаруживает и индицирует все случаи совпадения сетевых адресов у двух подключенных устройств (в стандарте IEEE это требование не обязательно).

Прочие и специальные требования:

- простота подключения к другому связному оборудованию, в том числе к арендованным линиям, телефонным сетям, сетям передачи данных (общего и частного пользования);
- простота интерфейсов между ЛВС и подключенными устройствами;
- защита передаваемых данных от случайного или несанкционированного доступа;

- наличие средств сопряжения с другими ЛВС (мосты) или с большими сетями (шлюзы);
- обеспечение беспроводной связи для мобильных устройств;
- интеграция передачи по сети различных видов информации (данных, речи, изображений и др.).

1.4. Базовая модель взаимодействия систем

Главная цель, которая преследуется при соединении компьютеров в сеть, – это возможность совместного использования ресурсов каждого компьютера. Для того чтобы реализовать эту возможность, компьютеры, подсоединенные к сети, должны иметь необходимые для этого средства взаимодействия с другими компьютерами сети. Задача разделения сетевых ресурсов является сложной, она включает в себя решение множества проблем: выбор способа адресации компьютеров и согласование электрических сигналов при установлении электрической связи, обеспечение надежной передачи данных и обработка сообщений об ошибках, формирование отправляемых и интерпретация полученных сообщений, а также много других не менее важных задач.

Обычным подходом при решении сложной проблемы является ее декомпозиция на несколько частных проблем – подзадач. Для решения каждой подзадачи назначается некоторый модуль. При этом четко определяются функции каждого модуля и правила их взаимодействия.

Частным случаем декомпозиции задачи является многоуровневое представление, при котором все множество модулей, решающих подзадачи, разбивается на иерархически упорядоченные группы – уровни. Для каждого уровня определяется набор функций-запросов, с которыми к модулям данного уровня могут обращаться модули вышележащего уровня для решения своих задач. Такой формально определенный набор функций, выполняемых данным уровнем для вышележащего уровня, а также форматы сообщений, которыми обмениваются два соседних уровня в ходе своего взаимодействия, называется *интерфейсом*.

Интерфейс определяет совокупный сервис, предоставляемый данным уровнем вышележащему уровню.

При организации взаимодействия компьютеров в сети каждый уровень ведет "переговоры" с соответствующим уровнем другого компьютера.

При передаче сообщений оба участника сетевого обмена должны принять множество соглашений. Например, они должны согласовать уровни и форму электрических сигналов, способ определения длины сообщений, договориться о методах контроля достоверности и т.п. Другими словами, соглашения должны быть приняты для всех уровней, начиная от самого низкого уровня передачи битов до самого высокого уровня, детализирующего, как информация должна быть интерпретирована.

Правила взаимодействия двух машин могут быть описаны в виде набора процедур для каждого из уровней. Такие формализованные правила, определяющие последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на одном уровне, но в разных узлах, называются *протоколами*.

Из приведенных определений можно заметить, что понятия "интерфейс" и "протокол", в сущности, обозначают одно и то же, а именно формализованно заданные процедуры взаимодействия компонент, решающих задачу связи компьютеров в сети. Однако довольно часто в использовании этих терминов имеется некоторый нюанс: понятие "протокол" чаще применяют при описании правил взаимодействия компонентов одного уровня, расположенных на разных узлах сети, а "интерфейс" – при описании правил взаимодействия компонентов соседних уровней, расположенных в пределах одного узла.

Согласованный набор протоколов разных уровней, достаточный для организации межсетевого взаимодействия, называется *стеком протоколов*.

Программные средства, реализующие некоторый протокол, также называют протоколом. При этом соотношение между протоколом – формально определенной процедурой взаимодействия и протоколом – средством, реализующим эту процедуру, аналогично соотношению между алгоритмом решения некоторой задачи и программой, решающей эту задачу. Понятно, что один и тот же алгоритм может быть запрограммирован с разной степенью эффективности. Точно так же и протокол может иметь несколько программных реализаций, например, протокол IPX, реализованный компанией Microsoft для Windows NT в виде программного продукта NWLink, имеет характеристики, отличающиеся от реализации этого же протокола компанией Novell. Именно поэтому, при сравнении протоколов следует учитывать не только логику их работы, но и качество программных решений. Более того, на эффективность взаимодействия устройств в сети влияет качество всей совокупности протоколов, составляющих стек,

то есть насколько рационально распределены функции между протоколами разных уровней и насколько хорошо определены интерфейсы между ними.

Протоколы реализуются не только программно-аппаратными средствами компьютеров, но и коммуникационными устройствами. Действительно, в общем случае связь компьютеров в сети осуществляется не напрямую "компьютер – компьютер", а через различные коммуникационные устройства, такие, например, как концентраторы, коммутаторы или маршрутизаторы. В зависимости от типа устройства в нем должны быть встроены средства, реализующие некоторый набор сетевых протоколов.

При организации взаимодействия могут быть использованы два основных типа протоколов. В протоколах *с установлением соединения* (connection-oriented network service – CONS) перед обменом данными отправитель и получатель должны сначала установить логическое соединение, то есть договориться о параметрах процедуры обмена, которые будут действовать только в рамках данного соединения. После завершения диалога они должны разорвать это соединение. Когда устанавливается новое соединение, переговорная процедура выполняется заново. Телефон – это пример взаимодействия, основанного на установлении соединения.

Вторая группа протоколов – протоколы *без предварительного установления соединения* (connectionless network service – CLNS). Такие протоколы называются также *дейтаграммными* протоколами. Отправитель просто передает сообщение, когда оно готово. Опускание письма в почтовый ящик – это пример связи без установления соединения.

Модель ISO/OSI

Из того, что протокол является соглашением, принятым двумя взаимодействующими объектами, в данном случае двумя работающими в сети компьютерами, совсем не следует, что он обязательно представляет собой стандарт. Но на практике при реализации сетей стремятся использовать стандартные протоколы. Это могут быть фирменные, национальные или международные стандарты.

Международная организация по стандартам (International Standards Organization – ISO) разработала модель, которая четко определяет различные уровни взаимодействия систем, дает им стандартные имена и указывает, какую работу должен делать каждый уровень. Эта модель называется

моделью взаимодействия открытых систем (Open System Interconnection – OSI) или моделью ISO/OSI.

В модели OSI взаимодействие делится на семь уровней или слоев (рис. 1.2). Каждый уровень имеет дело с одним определенным аспектом взаимодействия. Таким образом, проблема взаимодействия декомпозирована на 7 частных проблем, каждая из которых может быть решена независимо от других. Каждый уровень поддерживает интерфейсы с выше- и нижележащими уровнями.

Модель OSI описывает только системные средства взаимодействия, не касаясь приложений конечных пользователей. Приложения реализуют свои собственные протоколы взаимодействия, обращаясь к системным средствам. Следует иметь в виду, что приложение может взять на себя функции некоторых верхних уровней модели OSI, в таком случае при необходимости межсетевых обмена оно обращается напрямую к системным средствам, выполняющим функции оставшихся нижних уровней модели OSI.

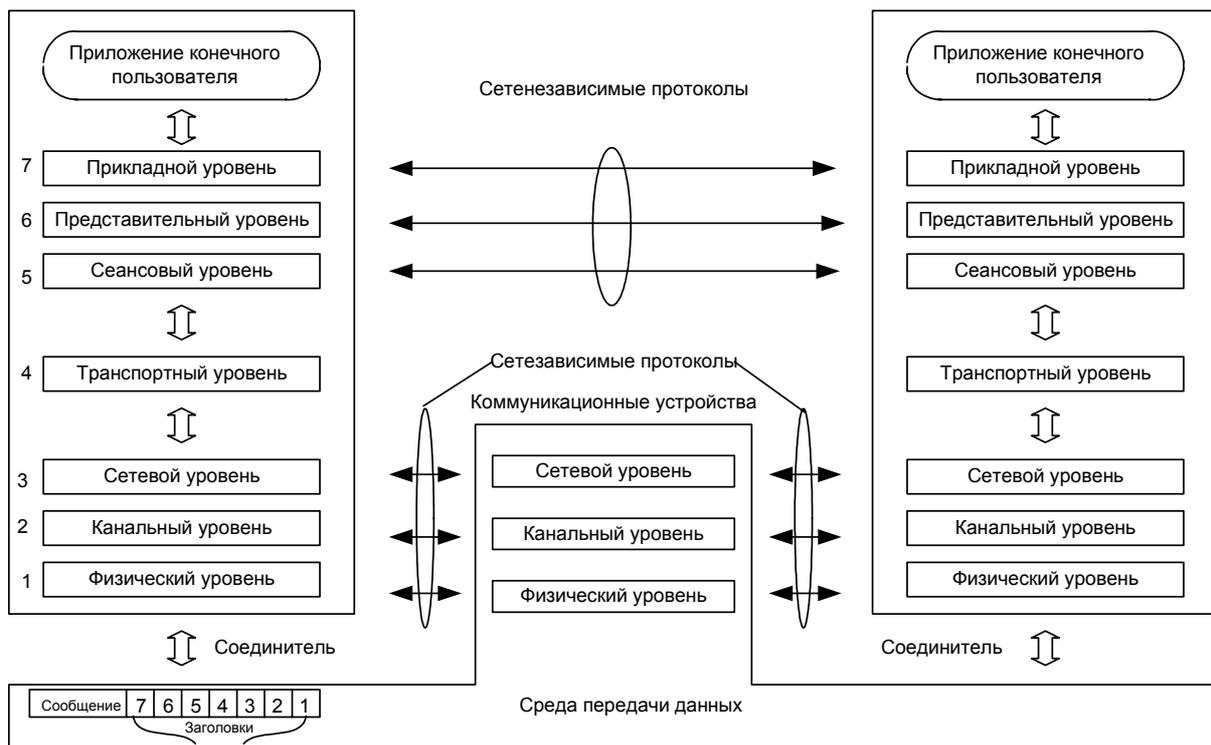


Рис. 1.2. Модель взаимодействия открытых систем ISO/OSI

Приложение конечного пользователя может использовать системные средства взаимодействия не только для организации диалога с другим при-

ложением, выполняющимся на другой машине, но и просто для получения услуг того или иного сетевого сервиса, например доступа к удаленным файлам, получение почты или печати на разделяемом принтере.

Итак, пусть приложение обращается с запросом к прикладному уровню, например к файловому сервису. На основании этого запроса программное обеспечение прикладного уровня формирует сообщение стандартного формата, в которое помещает служебную информацию (заголовок) и, возможно, передаваемые данные. Затем это сообщение направляется представителю уровня. Представительный уровень добавляет к сообщению свой заголовок и передает результат вниз сеансовому уровню, который, в свою очередь, добавляет свой заголовок и т.д. Некоторые реализации протоколов предусматривают наличие в сообщении не только заголовка, но и концевика. Наконец, сообщение достигает самого низкого, физического уровня, который действительно передает его по линиям связи.

Когда сообщение по сети поступает на другую машину, оно последовательно перемещается вверх с уровня на уровень. Каждый уровень анализирует, обрабатывает и удаляет заголовок своего уровня, выполняет соответствующие данному уровню функции и передает сообщение вышележащему уровню.

Кроме термина "сообщение" (message) существуют и другие названия, используемые сетевыми специалистами для обозначения единицы обмена данными. В стандартах ISO для протоколов любого уровня используется такой термин, как "протокольный блок данных" (Protocol Data Unit – PDU). Кроме этого часто используются названия «кадр» (frame), «пакет» (packet), «дейтаграмма» (datagram).

Функции уровней модели ISO/OSI

Физический уровень. Этот уровень имеет дело с передачей битов по физическим каналам, таким, например, как коаксиальный кабель, витая пара или оптоволоконный кабель. К этому уровню имеют отношение характеристики физических сред передачи данных, такие как полоса пропускания, помехозащищенность, волновое сопротивление и другие. На этом же уровне определяются характеристики электрических сигналов: требования к фронтам импульсов, уровням напряжения или тока передаваемого

сигнала, тип кодирования, скорость передачи сигналов. Кроме этого здесь стандартизируются типы разъемов и назначение каждого контакта.

Функции физического уровня реализуются во всех устройствах, подключенных к сети. Со стороны компьютера функции физического уровня выполняются сетевым адаптером или последовательным портом.

Примером протокола физического уровня может служить спецификация 10Base-T технологии Ethernet, которая определяет в качестве используемого кабеля неэкранированную витую пару категории 3 с волновым сопротивлением 100 Ом, разъем RJ-45, максимальную длину физического сегмента 100 м, манчестерский код для представления данных на кабеле и другие характеристики среды и электрических сигналов.

Канальный уровень. На физическом уровне просто пересылаются биты. При этом не учитывается, что в некоторых сетях, в которых линии связи используются (разделяются) попеременно несколькими парами взаимодействующих компьютеров, физическая среда передачи может быть занята. Поэтому одной из задач канального уровня является проверка доступности среды передачи. Другой задачей канального уровня является реализация механизмов обнаружения и коррекции ошибок. Для этого на канальном уровне биты группируются в наборы, называемые кадрами (frames). Канальный уровень обеспечивает корректность передачи каждого кадра, помещая специальную последовательность бит в начало и конец каждого кадра, чтобы отметить его, а также вычисляет контрольную сумму, суммируя все байты кадра определенным способом и добавляя контрольную сумму к кадру. Когда кадр приходит, получатель снова вычисляет контрольную сумму полученных данных и сравнивает результат с контрольной суммой из кадра. Если они совпадают, кадр считается правильным и принимается. Если же контрольные суммы не совпадают, то фиксируется ошибка.

В протоколах канального уровня, используемых в локальных сетях, заложена определенная структура связей между компьютерами и способы их адресации. Хотя канальный уровень и обеспечивает доставку кадра между любыми двумя узлами локальной сети, он это делает только в сети с совершенно определенной топологией связей, именно той топологией, для которой он был разработан. К таким типовым топологиям, поддерживаемым протоколами канального уровня локальных сетей, относятся «общая

шина», «кольцо» и «звезда». Примерами протоколов канального уровня являются протоколы Ethernet, Token Ring, FDDI, 100VG-AnyLAN.

В локальных сетях протоколы канального уровня используются компьютерами, мостами, коммутаторами и маршрутизаторами. В компьютерах функции канального уровня реализуются совместными усилиями сетевых адаптеров и их драйверов.

В глобальных сетях, которые редко обладают регулярной топологией, канальный уровень обеспечивает обмен сообщениями между двумя соседними компьютерами, соединенными индивидуальной линией связи. Примерами протоколов "точка – точка" (как часто называют такие протоколы) могут служить широко распространенные протоколы PPP и LAP-B.

Сетевой уровень. Этот уровень служит для образования единой транспортной системы, объединяющей несколько сетей с различными принципами передачи информации между конечными узлами. Рассмотрим функции сетевого уровня на примере локальных сетей. Протокол канального уровня локальных сетей обеспечивает доставку данных между любыми узлами только в сети с соответствующей *типовой топологией*. Это очень жесткое ограничение, которое не позволяет строить сети с развитой структурой, например, сети, объединяющие несколько сетей предприятия в единую сеть, или высоконадежные сети, в которых существуют избыточные связи между узлами. Для того чтобы, с одной стороны, сохранить простоту процедур передачи данных для типовых топологий, а с другой стороны, допустить использование произвольных топологий, используется дополнительный сетевой уровень. На этом уровне вводится понятие "сеть". В данном случае под сетью понимается совокупность компьютеров, соединенных между собой в соответствии с одной из стандартных типовых топологий и использующих для передачи данных один из протоколов канального уровня, определенный для этой топологии.

Таким образом, внутри сети доставка данных регулируется канальным уровнем, а вот доставкой данных между сетями занимается сетевой уровень.

Сообщения сетевого уровня принято называть *пакетами (packets)*. При организации доставки пакетов на сетевом уровне используется понятие "*номер сети*". В этом случае адрес получателя состоит из номера сети и номера компьютера в этой сети.

Сети соединяются между собой специальными устройствами, называемыми маршрутизаторами. *Маршрутизатор* – это устройство, которое собирает информацию о топологии межсетевых соединений и на ее основании пересылает пакеты сетевого уровня в сеть назначения. Для того чтобы передать сообщение от отправителя, находящегося в одной сети, получателю, находящемуся в другой сети, нужно совершить некоторое количество транзитных передач (hops) между сетями, каждый раз выбирая подходящий маршрут. Таким образом, маршрут представляет собой последовательность маршрутизаторов, через которые проходит пакет.

Проблема выбора наилучшего пути называется *маршрутизацией* и ее решение является главной задачей сетевого уровня. Эта проблема осложняется тем, что самый короткий путь не всегда самый лучший. Часто критерием при выборе маршрута является время передачи данных по этому маршруту, оно зависит от пропускной способности каналов связи и интенсивности трафика, которая может изменяться с течением времени. Некоторые алгоритмы маршрутизации пытаются приспособиться к изменению нагрузки, в то время как другие принимают решения на основе средних показателей за длительное время. Выбор маршрута может осуществляться и по другим критериям, например надежности передачи.

На сетевом уровне определяется два вида протоколов. Первый вид относится к определению правил передачи пакетов с данными конечных узлов от узла к маршрутизатору и между маршрутизаторами. Именно эти протоколы обычно имеют в виду, когда говорят о протоколах сетевого уровня. К сетевому уровню относят и другой вид протоколов, называемых *протоколами обмена маршрутной информацией*. С помощью этих протоколов маршрутизаторы собирают информацию о топологии межсетевых соединений. Протоколы сетевого уровня реализуются программными модулями операционной системы, а также программными и аппаратными средствами маршрутизаторов.

Примерами протоколов сетевого уровня являются протокол межсетевого взаимодействия IP стека TCP/IP и протокол межсетевого обмена пакетами IPX стека Novell.

Транспортный уровень. На пути от отправителя к получателю пакеты могут быть искажены или утеряны. Хотя некоторые приложения имеют собственные средства обработки ошибок, существуют и такие, которые

предпочитают сразу иметь дело с надежным соединением. Работа транспортного уровня заключается в том, чтобы обеспечить приложениям или верхним уровням стека – прикладному и сеансовому – передачу данных с той степенью надежности, которая им требуется. Модель OSI определяет пять классов сервиса, предоставляемых транспортным уровнем. Эти виды сервиса отличаются качеством предоставляемых услуг: срочностью, возможностью восстановления прерванной связи, наличием средств мультиплексирования нескольких соединений между различными прикладными протоколами через общий транспортный протокол, а главное – способностью к обнаружению и исправлению ошибок передачи, таких как искажение, потеря и дублирование пакетов.

Выбор класса сервиса транспортного уровня определяется, с одной стороны, тем, в какой степени задача обеспечения надежности решается самими приложениями и протоколами более высоких, чем транспортный, уровней, а с другой стороны, этот выбор зависит от того, насколько надежной является вся система транспортировки данных в сети. Так, например, если качество каналов передачи связи очень высокое и вероятность возникновения ошибок, не обнаруженных протоколами более низких уровней, невелика, то разумно воспользоваться одним из облегченных сервисов транспортного уровня, не обремененных многочисленными проверками, квитированием и другими приемами повышения надежности. Если же транспортные средства изначально очень ненадежны, то целесообразно обратиться к наиболее развитому сервису транспортного уровня, который работает, используя максимум средств для обнаружения и устранения ошибок: с помощью предварительного установления логического соединения, контроля доставки сообщений с помощью контрольных сумм и циклической нумерации пакетов, установления тайм-аутов доставки и т.п.

Как правило, все протоколы, начиная с транспортного уровня и выше, реализуются программными средствами конечных узлов сети – компонентами их сетевых операционных систем. В качестве примера транспортных протоколов можно привести протоколы TCP и UDP стека TCP/IP и протокол SPX стека Novell.

Сеансовый уровень. Сеансовый уровень обеспечивает управление диалогом для того, чтобы фиксировать, какая из сторон является активной

в настоящий момент, а также предоставляет средства синхронизации. Последние позволяют вставлять контрольные точки в длинные передачи, чтобы в случае отказа можно было вернуться назад к последней контрольной точке, вместо того, чтобы начинать все с начала. На практике немногие приложения используют сеансовый уровень, и он редко реализуется.

Уровень представления. Этот уровень обеспечивает гарантию того, что информация, передаваемая прикладным уровнем, будет понятна прикладному уровню в другой системе. При необходимости уровень представления выполняет преобразование форматов данных в некоторый общий формат представления, а на приеме соответственно выполняет обратное преобразование. Таким образом, прикладные уровни могут преодолеть, например, синтаксические различия в представлении данных. На этом уровне может выполняться шифрование и дешифрирование данных, благодаря которому секретность обмена данными обеспечивается сразу для всех прикладных серверов. Примером протокола, работающего на уровне представления, является протокол Secure Socket Layer (SSL), который обеспечивает секретный обмен сообщениями для протоколов прикладного уровня стека TCP/IP.

Прикладной уровень. Прикладной уровень – это в действительности просто набор разнообразных протоколов, с помощью которых пользователи сети получают доступ к разделяемым ресурсам, таким как файлы, принтеры или гипертекстовые Web-страницы, а также организуют свою совместную работу, например, с помощью протокола электронной почты. Единица данных, которой оперирует прикладной уровень, обычно называется *сообщением (message)*.

Существует очень большое разнообразие протоколов прикладного уровня. Приведем в качестве примеров хотя бы несколько наиболее распространенных реализаций файловых сервисов: NCP в операционной системе Novell NetWare, SMB в Microsoft Windows NT, NFS, FTP и TFTP, входящие в стек TCP/IP.

Модель OSI представляет хотя и очень важную, но только одну из многих моделей коммуникаций. Эти модели и связанные с ними стеки протоколов могут отличаться количеством уровней, их функциями, форматами сообщений, сервисами, предоставляемыми на верхних уровнях, и прочими параметрами.

Протоколы взаимодействия приложений и протоколы транспортной подсистемы

Функции всех уровней модели OSI могут быть отнесены к одной из двух групп: либо к функциям, зависящим от конкретной технической реализации сети, либо к функциям, ориентированным на работу с приложениями.

Три нижних уровня – физический, канальный и сетевой – являются сетезависимыми, то есть протоколы этих уровней тесно связаны с технической реализацией сети, с используемым коммуникационным оборудованием. Например, переход на оборудование FDDI означает полную смену протоколов физического и канального уровней во всех узлах сети.

Три верхних уровня – сеансовый, уровень представления и прикладной – ориентированы на приложения и мало зависят от технических особенностей построения сети. На протоколы этих уровней не влияют никакие изменения в топологии сети, замена оборудования или переход на другую сетевую технологию. Так, переход от Ethernet на высокоскоростную технологию АТМ не потребует никаких изменений в программных средствах, реализующих функции прикладного, представительного и сеансового уровней.

Транспортный уровень является промежуточным, он скрывает все детали функционирования нижних уровней от верхних уровней. Это позволяет разрабатывать приложения, не зависящие от технических средств, непосредственно занимающихся транспортировкой сообщений.

Рис. 1.2 показывает и уровни модели OSI, на которых работают различные элементы сети. Компьютер с установленной на нем сетевой ОС взаимодействует с другим компьютером с помощью протоколов всех семи уровней. Это взаимодействие компьютеры осуществляют через различные коммуникационные устройства: концентраторы, модемы, мосты, коммутаторы, маршрутизаторы, мультиплексоры. В зависимости от типа, коммуникационное устройство может работать либо только на физическом уровне (повторитель), либо на физическом и канальном (мост и коммутатор), либо на физическом, канальном и сетевом, иногда захватывая и транспортный уровень (маршрутизатор).

Особенности протоколов, используемых в локальных и глобальных сетях

В настоящее время наблюдается тенденция к сближению протоколов локальных и глобальных сетей. Ярким примером являются протоколы технологии АТМ, работающие без изменений как в тех, так и в других сетях. Тем не менее, большинство протоколов, используемых сегодня, относятся либо к локальным, либо к глобальным сетям и не могут применяться не по прямому назначению.

Отличия между протоколами локальных и глобальных сетей связаны с различными свойствами каналов, использующихся в этих сетях.

Каналы локальных сетей имеют небольшую длину и высокое качество, а каналы глобальных сетей – наоборот, большую длину и низкое качество.

Небольшая длина каналов локальных сетей создала возможность совместного использования их узлами сети в режиме разделения времени. Практически все протоколы локальных сетей имеют версию работы на разделяемых средах передачи данных, хотя более поздние протоколы (Fast Ethernet, Gigabit Ethernet) имеют также и версию работы на индивидуальных каналах в полнодуплексном режиме. Большая протяженность каналов глобальных сетей делает нерациональными любые процедуры разделения канала во времени, так как длительность этих процедур становится слишком большой. Поэтому каналы глобальных сетей используются всегда на индивидуальной основе как связи типа "точка – точка".

Высокое качество кабелей локальных сетей послужило причиной отказа от использования в протоколах локальных сетей процедур восстановления искаженных и потерянных кадров. Этих процедур нет ни в протоколах семейства Ethernet, ни у протокола Token Ring, ни у протокола FDDI. В то же время в протоколах глобальных сетей, ориентирующихся на каналы плохого качества, процедурам восстановления кадров всегда уделялось большое внимание. Например, в сетях X.25 восстановлением кадров занимаются сразу два смежных протокола: LAP-B – на канальном уровне, X.25 – на сетевом.

Начало массового использования цифровых оптоволоконных каналов в глобальных сетях, обеспечивающих высокое качество передачи данных, послужило причиной разработки протоколов глобальных сетей ново-

го поколения, в которых отсутствуют процедуры восстановления кадров. Такой особенностью обладают, например, сети frame relay и АТМ.

Таким образом, одно из отличий протоколов локальных и глобальных сетей преодолено за счет продвижения глобальных сетей навстречу локальным. Второе отличие сегодня снимается за счет быстрого внедрения в локальные сети техники микросегментации, отказывающейся от использования разделяемых сред и предоставляющей каждому узлу сети индивидуальный коммутируемый канал. В результате протоколы локальных и глобальных сетей все больше сближаются, а существование технологии АТМ доказывает, что принципиальных причин для существования четкой границы между этими классами протоколов сегодня не существует.

Контрольные вопросы

1. Какие типовые схемотехнические решения используются при построении вычислительной системы?
2. Что такое вычислительная система?
3. Назовите базовый состав оборудования, входящего в состав вычислительной системы.
4. Является ли понятие «интерфейс» определением для программных средств передачи данных?
5. Набор аппаратных средств сопряжения вычислительной системы с объектом относится к понятию «стык», «протокол» или «интерфейс»?
6. Какие требования предъявляются к ЛВС?
7. Что входит в общие требования, предъявляемые к ЛВС?
8. Что входит в требования, предъявляемые к взаимодействию устройств в ЛВС?
9. Какие задачи входят в разделение сетевых ресурсов?
10. Приведите понятия интерфейса и протокола.
11. Что такое модель OSI/ISO?
12. Перечислите уровни модели OSI/ISO.
13. Назовите функции каждого из уровней модели OSI/ISO.
14. Какие уровни являются сетезависимыми?
15. На какие уровни не влияют изменения в топологии сети?

Глава 2. КОНФИГУРАЦИЯ ЛВС

Конфигурация, или топология, ЛВС определяет взаимное размещение станций сети и способ их соединения. Существуют следующие топологии ЛВС: шинная, кольцевая, звездообразная, петлевая, древовидная, гибридная и полносвязная.

2.1. Шинная топология ЛВС

При шинной топологии среда передачи информации представляется в форме коммуникационного пути, доступного для всех рабочих станций, к которому они все должны быть подключены (рис. 2.1). Все рабочие станции могут непосредственно вступать в контакт с любой рабочей станцией, имеющейся в сети. Рабочие станции в любое время, без прерывания работы всей вычислительной сети, могут быть подключены к ней или отключены. Функционирование вычислительной сети не зависит от состояния отдельной рабочей станции.

В стандартной ситуации для шинной сети Ethernet часто используют тонкий кабель или Cheapernet-кабель с тройниковым соединителем. Отключение и особенно подключение к такой сети требуют разрыва шины, что вызывает нарушение циркулирующего потока информации и зависание системы.

Новые технологии предлагают пассивные штепсельные коробки, через которые можно отключать и/или подключать рабочие станции во время работы вычислительной сети.

Благодаря тому, что рабочие станции можно подключать без прерывания сетевых процессов и коммуникационной среды, очень легко прослушивать информацию, т.е. отвлекать информацию из коммуникационной среды.

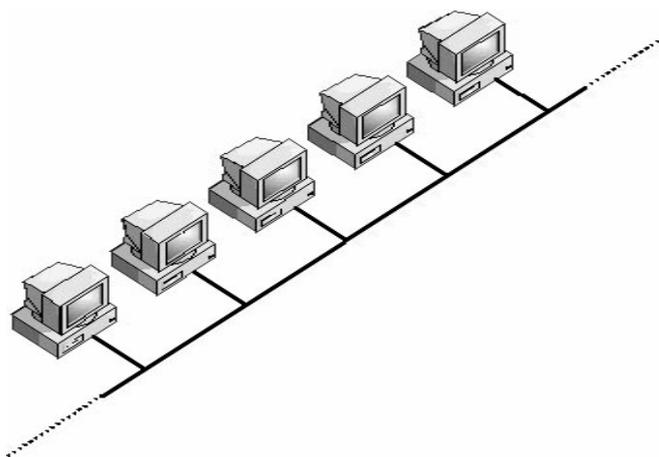


Рис. 2.1. Шинная топология

В ЛВС с прямой (немодулируемой) передачей информации всегда может существовать только одна станция, передающая информацию. Для предотвращения коллизий в большинстве случаев применяется временной метод разделения, согласно которому для каждой подключенной рабочей станции в определенные моменты времени предоставляется исключительное право на использование канала передачи данных. Поэтому требования к пропускной способности вычислительной сети при повышенной нагрузке повышаются, например, при вводе новых рабочих станций. Рабочие станции присоединяются к шине посредством устройств ТАР (Terminal Access Point – точка подключения терминала). ТАР представляет собой специальный тип подсоединения к коаксиальному кабелю. Зонд игольчатой формы внедряется через наружную оболочку внешнего проводника и слой диэлектрика к внутреннему проводнику и присоединяется к нему.

В ЛВС с модулированной широкополосной передачей информации различные рабочие станции получают, по мере надобности, частоту, на которой они могут отправлять и получать информацию. Пересылаемые данные модулируются на соответствующих несущих частотах, т.е. между средой передачи информации и рабочими станциями находятся соответственно модемы для модуляции и демодуляции. Техника широкополосных сообщений позволяет одновременно транспортировать в коммуникационной среде довольно большой объем информации. Для дальнейшего развития дискретной транспортировки данных не играет роли, какая первоначальная информация подана в модем (аналоговая или цифровая), так как она все равно в дальнейшем будет преобразована.

2.2. Кольцевая ЛВС

В ЛВС кольцевой конфигурации сигналы передаются по кольцу в большинстве случаев только в одном направлении, т.е. от рабочей станции 1 к рабочей станции 2, от рабочей станции 3 к рабочей станции 4 и т.д.; последняя рабочая станция связана с первой. Коммуникационная связь замыкается в кольцо (рис. 2.2). Каждая станция непосредственно подсоединяется только к двум соседним узлам и "прослушивает" передачу любой другой станции. Кольцо состоит из нескольких приемопередатчиков, соединенных физической средой. В кольцевой ЛВС может отсутствовать цен-

тральный управляющий узел и все станции имеют равные права доступа к физической среде. Однако во многих таких ЛВС одна из станций выполняет функции активного монитора, осуществляя инициацию, тестирование кольца, обнаружение и удаление искаженных или дублированных пакетов.

Прокладка кабелей от одной рабочей станции до другой может быть довольно сложной и дорогостоящей, особенно если географическое расположение рабочих станций далеко от формы кольца (например, в линию).

Сообщения циркулируют регулярно по кругу. Рабочая станция посылает по определенному конечному адресу информацию, предварительно получив из кольца запрос. Пересылка сообщений является очень эффективной, так как большинство сообщений можно отправлять «в дорогу» по кабельной системе одно за другим. Очень просто можно сделать кольцевой запрос на все станции. Продолжительность передачи информации увеличивается пропорционально количеству рабочих станций, входящих в вычислительную сеть.

Основная проблема при кольцевой топологии заключается в том, что каждая рабочая станция должна активно участвовать в пересылке информации и в случае выхода из строя хотя бы одной из них вся сеть парализуется. Неисправности в кабельных соединениях локализуются легко.

Подключение новой рабочей станции требует краткосрочного выключения сети, так как во время установки кольцо должно быть разомкнуто. Ограничения на протяженность вычислительной сети не существует, так как оно в конечном счете определяется исключительно расстоянием между двумя рабочими станциями.

Специальной формой кольцевой топологии является логическая кольцевая сеть (рис. 2.3). Физически она монтируется как соединение звездных топологий. Отдельные звезды включаются с помощью специальных коммутаторов (Hub – концентратор). В зависимости от числа рабочих станций и длины кабеля между рабочими станциями применяют активные

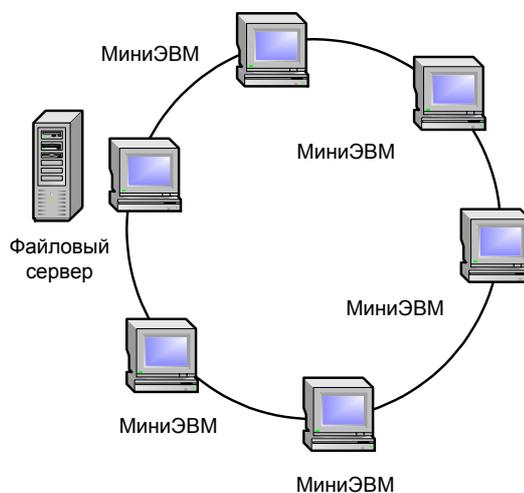


Рис. 2.2. Структура кольцевой топологии ЛВС

или пассивные концентраторы. Активные концентраторы дополнительно содержат усилитель для подключения от 4 до 16 рабочих станций. Пассивный концентратор является исключительно разветвляющим устройством (максимум на три рабочие станции). Управление отдельной рабочей станцией в логической кольцевой сети происходит так же, как и в обычной кольцевой сети. Каждой рабочей станции присваивается соответствующий ей адрес, по которому передается управление (от старшего к младшему и от самого младшего к самому старшему). Разрыв соединения происходит только для нижерасположенного (ближайшего) узла вычислительной сети, так что лишь в редких случаях может нарушаться работа всей сети.

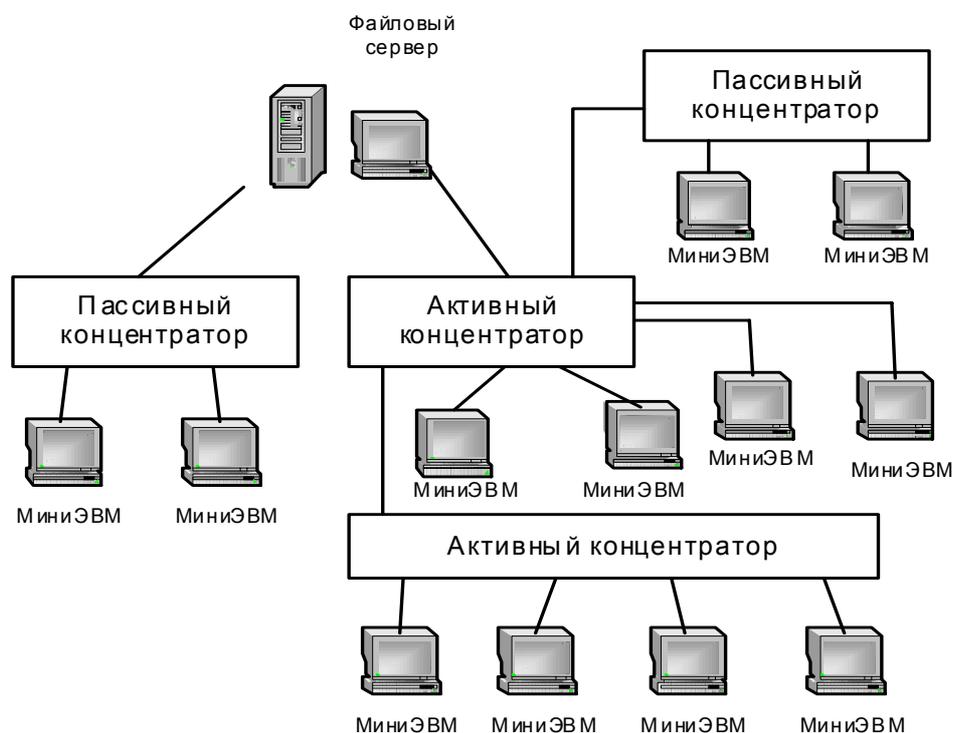


Рис. 2.3. Структура логической кольцевой сети ЛВС

2.3. Звездообразная конфигурация ЛВС

Концепция топологии сети в виде звезды пришла из области больших ЭВМ, где головная машина получает и обрабатывает все данные с периферийных устройств как активный узел обработки данных (рис. 2.4). Этот принцип применяется в системах передачи данных, например в элек-

36

тронной почте сети RelCom. Вся информация между двумя периферийными рабочими местами проходит через центральный узел вычислительной сети. Такая конфигурация характерна также и для обычных вычислительных сетей с терминальными устройствами или систем телеобработки данных. В ЛВС в центре звезды находится либо пассивный коммутатор, либо активное устройство, которое последовательно опрашивает станции и предоставляет им права на обмен данными. Традиционным методом работы такого типа ЛВС является коммутация каналов, хотя в последние годы в них начал широко использоваться метод коммутации пакетов. Центральный узел такой ЛВС действует обычно не только как коммутатор, но и как преобразователь скоростей и протоколов.

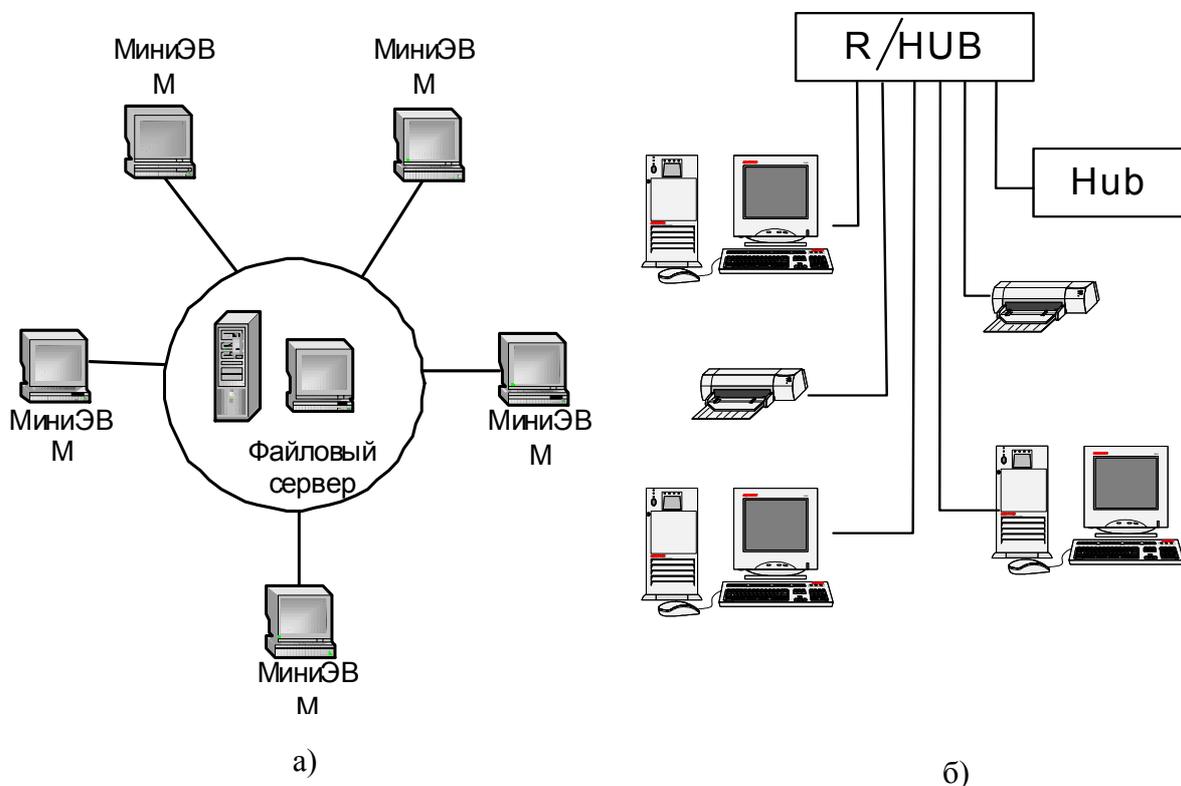


Рис. 2.4. Топология «звезда»

Каждый компьютер, как и любое устройство, способное устанавливать связь, подключен отдельным проводом к отдельному порту устройства, называемого концентратором или повторителем (repeater), или хабом (Hub).

Концентраторы могут быть как активные, так и (теоретически) пассивные. Если между устройством и концентратором происходит разрыв соединения, то вся остальная сеть продолжает работать. Правда, если этим устройством был единственный сервер, то работа будет несколько затруднена. При выходе из строя концентратора сеть перестанет работать. Данная сетевая топология наиболее удобна при поиске повреждений сетевых элементов: кабеля, сетевых адаптеров или разъемов. При добавлении новых устройств "звезда" также удобней по сравнению с топологией «Общая шина». Также можно принять во внимание, что 100- и 1000- Мегабитные сети строятся по топологии "звезда".

Пропускная способность сети определяется вычислительной мощностью узла и гарантируется для каждой рабочей станции. Коллизий (столкновений) данных не возникает. Кабельное соединение довольно простое, так как каждая рабочая станция связана с узлом. Затраты на прокладку кабелей высокие, особенно когда центральный узел географически расположен не в центре топологии.

2.4. Древоподобные ЛВС

Наряду с известными топологиями вычислительных сетей «кольцо», «звезда» и «шина», на практике применяется и комбинированная, например древоподобная, структура (рис. 2.5). Она образуется в основном в виде комбинаций вышеназванных топологий вычислительных сетей. Основание дерева вычислительной сети (корень) располагается в точке, в которой собираются коммуникационные линии информации (ветви дерева).

Вычислительные сети с древоподобной структурой применяются там, где невозможно непосредственное применение базовых сетевых структур в чистом виде. Для подключения большого числа рабочих станций применяют сетевые усилители и/или коммутаторы. Коммутатор, обладающий одновременно и функциями усилителя, называют активным концентратором. На практике применяют две их разновидности, обеспечивающие подключение соответственно восьми или шестнадцати линий.

Устройство, к которому можно присоединить максимум три станции, называют пассивным концентратором. Пассивный концентратор обычно используют как разветвитель. Он не нуждается в усилителе. Предпосылкой

для подключения пассивного концентратора является то, что возможное максимальное расстояние до рабочей станции не должно превышать нескольких десятков метров.

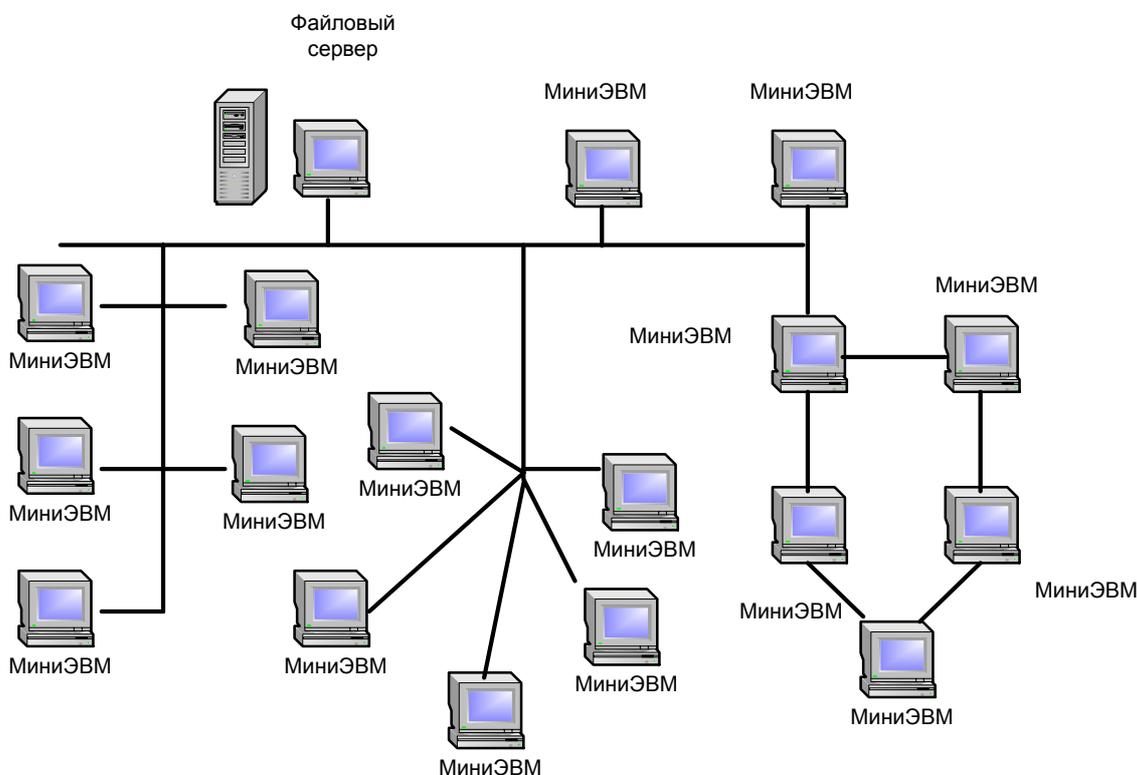


Рис. 2.5. Древоподобная структура ЛВС

2.5. Петлевые и полносвязные конфигурации

Петлевые ЛВС возникли как развитие многопунктовых линий с опросом. Но по своей конфигурации они сходны с кольцевыми ЛВС, differing from them by the method of distribution of access to the physical medium. In a ring network, there is a controlling station (or controller), which determines which specific station and for what purposes can use the physical medium. This is achieved by cyclic polling of all stations or by sending empty packets-containers, accessible to any station.

Полносвязные конфигурации обеспечивают выбор наиболее дешевого маршрута между абонентами и выгодны там, где усложнение логики управления окупается удешевлением связей.

2.6. Качественные характеристики ЛВС

По способу адресации и передачи данных все рассмотренные конфигурации можно сгруппировать в два основных класса ЛВС: широковещательные и последовательные.

В широковещательных конфигурациях каждая станция передает данные, которые могут быть восприняты всеми остальными станциями. К этому классу относятся шинная, древовидная и звездообразная конфигурации с пассивным коммутатором. В последовательных конфигурациях каждая станция передает данные только одной из станций. К этому классу относятся кольцевая, петлевая, звездообразная конфигурация с активным центром и сети смешанного типа.

Основные качественные характеристики конфигураций ЛВС приведены в таблице.

Качественные характеристики конфигураций ЛВС

Конфигурация	Достоинства	Недостатки
Шинная с временным разделением	Простота подключения новых станций и доступ ко всем компонентам сети; возможность эффективного использования пропускной способности сети; приспособленность к резким изменениям объема нагрузки; возможность подключения нескольких низкоскоростных устройств через один интерфейсный модуль; полностью пассивная физическая среда, не требующая электропитания; простота монтажа сети (благодаря отсутствию проблем маршрутизации)	Необходимость сложного интеллектуального устройства для связи со средой и подключения обычных терминалов; возможность взаимного наложения передаваемых по шине сообщений; отсутствие автоматического подтверждения приема; отсутствие контроля за справедливым распределением ресурсов; возможность необнаруживаемого несанкционированного прослушивания передачи; ограниченность общей длины шины

Конфигурация	Достоинства	Недостатки
Шинная с частотным разделением	Доступность кабеля и интерфейсных устройств; простота установки и трассировки кабеля, подключения новых ответвлений, новых станций и следовательно развития сети; приспособленность сети к продолжительной высокоскоростной передаче данных; возможность совмещенной передачи данных, изображений, речи и так далее по одному кабелю; возможность охвата большой территории	Необходимость использования дорогостоящих модемов; необходимость в большинстве применений постоянно функционирующего ретранслятора на конце шины; необходимость надежного снабжения электроэнергией линейных усилителей или повторителей
Кольцевая	Простота маршрутизации, организация широкополосной передачи и разделения пропускной способности между пользователями; отсутствие зависимости от центрального устройства; низкая стоимость повторителей (при однонаправленной передаче); простота идентификации неисправных участков; низкая вероятность ошибок передачи, простота их обнаружения и автоматическое подтверждение приема; гарантированный доступ к среде даже в сильно загруженной сети; возможность высоких скоростей передачи и использования комбинированной физической среды	Зависимость надежности сети от всех кабелей и повторителей; необходимость в большинстве случаев наличия мониторинговой станции; сложность удлинения кольца и подключения новых станций без прерывания функционирования кольца; задержка сигналов повторителями; необходимость близкого взаимного расположения повторителей и, следовательно, наличия большого их числа; сложность трассировки кабеля в некоторых случаях.

Конфигурация	Достоинства	Недостатки
Звездообразная	Простота доступа многих абонентов к одному центру обслуживания, возможность обслуживания простейших терминалов; возможность использования в разных направлениях различных типов каналов и скоростей передачи; простота обнаружения и устранения неисправностей; высокий уровень защиты от несанкционированного доступа к данным; простота адресации, контролируемой из центра; возможность интегрированной передачи данных и речи (интегрированной обработки учрежденческой информации)	Зависимость работоспособности сети от надежности центрального узла; сложность технологии и высокая стоимость центрального узла; необходимость портов в центральном узле для каждой линии или группы линий; пониженная (в сравнении с шинной и кольцевой топологиями) интенсивность потоков входных данных; высокие затраты на кабельные соединения
Древовидная	То же, что и при шинной топологии с временным разделением, и, кроме того, большие возможности изменения конфигурации и наращивания сети целыми группами станций	То же, что и при шинной топологии, кроме того, сложность управления при передаче немодулированных сигналов; необходимость тщательного подбора кабельных ответвлений с согласованными параметрами; более низкая скорость по сравнению с однокабельной шиной
Петлевая	Удобство взаимосвязи устройств с малыми вычислительными возможностями; низкая стоимость прокладки кабеля; использование известных процедур управления взаимодействием с ЭВМ; простота подключения новых станций	Зависимость функционирования сети от управляющего узла; невысокие скорости передачи данных; взаимодействие между станциями только через управляющий узел

Контрольные вопросы

1. Что такое топология сети и какие топологии существуют?
2. Шинная топология: способ организации связей, преимущества и недостатки.
3. Кольцевая топология: структура логической кольцевой сети.
4. Звездообразная конфигурация примеры реализации.
5. Достоинства и недостатки древовидных топологий.
6. Полносвязанная конфигурация и назначение управляющей станции.
7. Приведите сравнительные характеристики конфигураций по надежности и производительности.
8. Что общего и в чем отличие между взаимодействием компьютеров в сети и взаимодействием компьютера с периферийным устройством?
9. Поясните значения терминов «клиент», «сервер», «редиректор».
10. Назовите главные недостатки полносвязной топологии, а также топологий типа «общая шина», «звезда», «кольцо».
11. Какую топологию имеет односегментная сеть Ethernet, построенная на основе концентратора: «общая шина» или «звезда»?
12. В чем отличие логической структуризации сети от физической?
13. Какие из следующих утверждений верны:
 - а) разделение линий связи приводит к повышению пропускной способности канала;
 - б) конфигурация физических связей может совпадать с конфигурацией логических связей;
 - в) главной задачей службы разрешения имен является проверка сетевых имен и адресов на допустимость;
 - д) протоколы без установления соединений называются также дейтаграммными протоколами.

Глава 3. ФИЗИЧЕСКАЯ РЕАЛИЗАЦИЯ СРЕДЫ ПЕРЕДАЧИ ДАННЫХ

3.1. Общая характеристика каналов связи

Физически среда передачи данных представляет собой проводник (проводники), по которому передается информация с использованием различных видов кабелей (медная витая пара, коаксиальные, волоконно-оптические), а также эфир (радио-, микроволновые, инфракрасные каналы). В табл. 3.1 приведены основные параметры наиболее распространенных видов физической среды.

Таблица 3.1

Сравнительные характеристики физических сред ЛВС

Характеристика	Тип физической среды			
	Симметричный кабель	Коаксиальный кабель	Волоконно-оптический кабель	Эфир
Скорость передачи, Мбит/с	До 10	До 140	До 560	До 20000
Дальность передачи по одному сегменту, км	0,01...0,1	До 2,5	До 200	До 20
Типичное число узлов в сети	10...100	До 100	2	–
Сложность соединения	Низкая	Средняя	Очень высокая	Низкая
Возможность ответвления	Плохая	Средняя (100 узлов)	Плохая	Отличная
Возможность передачи различных видов информации	Низкая	Ограниченная	Очень хорошая	Очень хорошая
Помехозащищенность	Средняя	Высокая	Очень высокая	Высокая
Относительная стоимость (1м)	15	10	50	–

Симметричный кабель состоит из оболочки (с экраном или без), внутри которой содержится одна или несколько пар проводников. Симметричные кабели используются для телефонной связи и при подключениях телексных терминалов. При использовании на АТС электронных коммутаторов с цифровой передачей речи возможно построение ЛВС на имеющихся каналах. В ЛВС симметричные кабели применяются, как правило, в режиме передачи немодулированных сигналов, причем две или более пары проводников отводятся для передачи сигналов оповещения о предстоящей передаче данных. До недавнего времени одним из недостатков симметричных однопарных кабелей являлась низкая скорость передачи (до 1 Мбит/с). В настоящее время на таких кабелях достигнута скорость 10 Мбит/с и ожидается, что она может быть увеличена еще почти на порядок.

Симметричные многопарные кабели применяются в целом ряде ЛВС. Отдельные провода кабеля могут использоваться для разных целей (передача данных, сигналов идентификации, индикация состояний и др.). Передача данных по нескольким параллельным линиям умножает пропускную способность всего кабеля (сотни мегабит в секунду) при сравнительно малой скорости передачи сигналов по одному проводу (десятки мегабит в секунду). Низкая скорость передачи сигналов снимает проблемы отражения сигналов и согласования импедансов, характерные для высокоскоростных линий, упрощает и удешевляет интерфейсные схемы, хотя и увеличивает их необходимое число.

Основными недостатками симметричных кабелей остаются простота несанкционированного доступа и чувствительность к электромагнитным помехам (при отсутствии экрана). Поэтому симметричные кабели применяются главным образом в кольцевых сетях, где используются повторители, имеется возможность стыковать различные типы кабелей и вставлять в критическом месте нечувствительную к помехам секцию кабеля.

Коаксиальный кабель состоит из внутреннего проводника, окруженного слоем изолирующего материала, внешнего проводника и оболочки. Существует много разновидностей коаксиального кабеля с разными характеристиками. Коаксиальные кабели отличаются широкой полосой пропускания, обладают меньшим затуханием, более высокой устойчивостью к наводкам и т.д. Высококачественный кабель обладает большой жесткостью, и его трудно монтировать. Кроме того, электрические характеристики коаксиального кабеля (например, его волновое сопротивление в рабочем диапазоне частот составляет 50...75 Ом/м) делают его неудобным

для многих целей. Но он очень удобен для передачи высокочастотных сигналов при сохранении относительной устойчивости к электрическим наводкам, а также для передачи модулированных и немодулированных сигналов. Кабель отличается надежностью, простотой конструкции и умеренной массой. В сетях кабельного телевидения используется коаксиальный кабель с полосой пропускания более 300 МГц, обеспечивающий передачу сигнала на большие расстояния. В режиме передачи немодулированных сигналов коаксиальный кабель позволяет передавать информацию со скоростью 10 Мбит/с. Эти свойства коаксиального кабеля обусловили его использование в качестве физической среды большинства ЛВС.

В некоторых экспериментальных системах, системах коммунального назначения (например, для дистанционного управления освещением) используются силовые кабели, общедоступные и всегда готовые к эксплуатации. Однако низкая скорость передачи, сложность регулирования, проблемы безопасности и возможность воздействия со стороны других подобных систем ограничивают их применение в ЛВС.

В ряде систем (например, в Series I Ring (CM/1) фирмы IBM) применяются биаксиальные и триаксиальные кабели, которые имеют лучшие электрические характеристики, чем коаксиальный кабель.

Волоконно-оптический кабель (световод) основан на использовании в качестве проводящей среды сверхпрозрачного стекловолокна. Теоретический предел пропускной способности световода определяется сотнями гигабит в секунду, а на практике уже достигнута скорость 2,41 Гбит/с [5].

Помимо высокой скорости передачи к достоинствам световода следует отнести его высокую помехозащищенность, защищенность от несанкционированного доступа и небольшую массу. К его недостаткам относятся высокая стоимость и сложность подключения новых станций из-за неразвитости технологии волоконно-оптических разветвителей, по световодам нельзя передавать электрическую энергию для повторителей и других устройств. Подключение ответвителей вызывает значительное ослабление сигналов. Сигналы могут передаваться по кабелю только в одном направлении. Ожидается, однако, что многие из перечисленных проблем будут решаться по мере развития технологии волоконной оптики и расширения ее применения в ЛВС.

Световоды применяются в ЛВС кольцевой и звездообразной конфигурациях, например в ЛВС Fibernet фирмы Xerox, имеющей топологию звезды. Они наиболее подходят для взаимосвязей больших ЭВМ, где тре-

буются высокие скорости передачи. В то же время их применение для работы при средних и низких скоростях не считается целесообразным.

Новым типом физической среды для ЛВС является эфир, в котором могут быть организованы радиоканалы, инфракрасные каналы и микроволновые каналы.

Радиоканал наиболее пригоден для обслуживания мобильных станций. В стационарных ЛВС радиоканалы используются редко из-за экранированности зданий и узкой полосы доступных радиочастот.

Инфракрасный канал обеспечивает высокие скорости передачи (несколько мегабит в секунду) на расстояние прямой видимости. В отличие от радиоканалов он нечувствителен к электромагнитным помехам и не занимает полосы частот радио- или видеосигналов. К недостаткам инфракрасного канала следует отнести небольшую дальность передачи.

Микроволновый канал обеспечивает еще более высокие скорости (до 20 Гбит/с) на расстояние 15...20 км (при обеспечении прямой видимости).

3.2. Витая пара

Общие положения и классификация кабеля

"Витая пара" (twisted pair) – это кабель на медной основе, объединяющий в оболочке одну или более пар проводников. Каждая пара представляет собой два перевитых вокруг друг друга изолированных медных провода и может иметь несколько дополнительных защитных, экранирующих и технологических элементов, которые образуют сердечник. Каждый провод снабжается изоляцией из сплошного или вспененного диэлектрика. Использование последнего несколько снижает удельную массу кабеля и значительно улучшает его частотные свойства, однако приводит к удорожанию готового изделия.

На сердечник наложена защитная оболочка в виде шланга, в большем или меньшем объеме предохраняющая витые пары от внешних воздействий и сохраняющая структуру сердечника во время прокладки и эксплуатации.

В зависимости от основной области применения и соответственно конструкции кабельные изделия на основе витых пар подразделяются на четыре основных вида:

- горизонтальный кабель;
- магистральный кабель;

- кабель для шнуров;
- провод для переключателей.

Кабели СКС должны отвечать требованиям пожарной безопасности. На основе кабелей "витая пара" могут быть реализованы все три подсистемы СКС, хотя на внешних магистралях их применение для высокоскоростных приложений класса D затруднено ввиду достаточно жестких физических ограничений на максимальную длину сегмента.

На основании этого большинство электрических кабелей предназначено для применения внутри здания.

Имеется также ограниченная номенклатура кабелей на основе витых пар, которые могут прокладываться между зданиями (так называемые уличные кабели или outdoor-кабели).

Горизонтальный кабель типа "витая пара", конструктивные особенности. Горизонтальный кабель типа "витая пара" предназначен для использования в горизонтальной подсистеме на участке от коммутационного оборудования этажа до информационных розеток рабочих мест.

Наиболее распространенные на практике конструкции содержат четыре витые пары. По видам скрутки проводников горизонтального кабеля различают парную и четверочную витые пары (рис. 3.1).

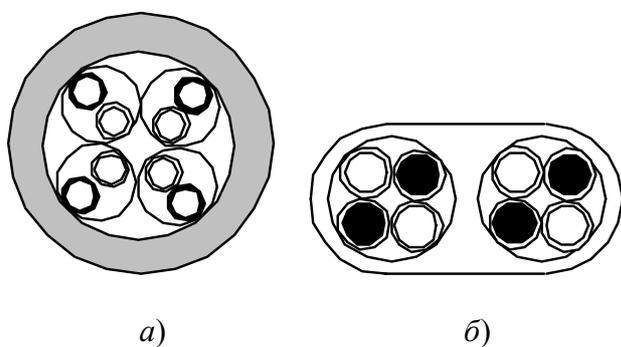


Рис. 3.1. Виды скруток витых пар:
а – парная; б – четверочная

Четверочная скрутка позволяет добиться меньших внешних габаритов, большей стабильности его конструкции и лучших электрических характеристик, однако кабель с четверочной скруткой более сложен в производстве и разделке и поэтому достаточно мало распространен.

В качестве материала изоляции проводников обычно используется поливинилхлорид, встречаются также другие изоляционные материалы, например: полиолефин, полиэтилен и полипропилен.

Применяются как сплошные, так и вспененные материалы, причем последние позволяют получить несколько лучшие электрические характе-

ристики, однако являются более дорогими и используются преимущественно в кабелях с верхней граничной частотой выше 100 МГц.

С целью снижения уровня затухания проводники горизонтального кабеля изготавливаются из монолитной (solid) медной проволоки. Отдельные витые пары образуют кабельный сердечник, покрытый общей для всех пар внешней защитной изоляционной оболочкой толщиной примерно 0,5 – 0,6 мм.

Для придания сердечнику определенной структуры в процессе производства и ее сохранения во время эксплуатации может применяться обмотка пар полимерными ленточками или нитями.

Облегчение разделки некоторых конструкций кабелей обеспечивается использованием разрывной нити (rip-cord), расположенной под оболочкой. При вытягивании эта нить делает на оболочке продольный разрез и открывает доступ к кабельному сердечнику.

Кабели "витая пара", у которых под общей оболочкой находятся три и более четырех парных элемента, относятся к многопарным. Для изготовления внешней оболочки наряду с обычным поливинилхлоридом достаточно часто применяется материал типа компаунда, который не содержит галогенов и не поддерживает горения, а также так называемые малоуглеродные полимеры.

Полному вытеснению поливинилхлорида из материалов оболочки препятствует тот факт, что переход на оболочку из негорючих материалов немедленно увеличивает цену готового продукта примерно на 20 – 30 %, а не содержащие галогенов компаунды обладают низкой огнестойкостью.

Внешняя оболочка окрашивается обычно в серый цвет различных оттенков, встречаются также другие стандартные для конкретного производителя цвета (синий, фиолетовый, белый, красный). Оранжевая окраска обычно указывает на то, что оболочка изготовлена из негорючего материала и кабель может быть использован для прокладки в так называемых *plenum*-полостях.

Конструкции, предназначенные для внешней прокладки, снабжаются полиэтиленовой оболочкой, так как этот материал обладает существенно

более высокой влагостойкостью по сравнению с поливинилхлоридом и огнестойким компаундом.

На внешнюю оболочку наносятся маркирующие надписи, в которых указывается тип кабеля, диаметр и тип проводников, характеристики и оболочки, наименование производителя и его фирменное обозначение кабеля, наименование стандарта и сертифицирующей лаборатории, а также футовые или метровые метки длины.

По двум последним параметрам имеются определенные различия между американскими и европейскими кабельными компаниями. Так, основной сертифицирующей лабораторией для американских производителей кабельной продукции является UL Laboratory, европейские обращаются в датскую испытательную организацию DELTA. Американские кабельные компании применяют в основном футовые метки длины, европейские изготовители используют метровый дискрет этого параметра.

Экранированный и неэкранированный горизонтальные кабели типа "витая пара". В зависимости от наличия или отсутствия дополнительных экранирующих покрытий отдельных витых пар и/или сердечника в целом горизонтальные кабели из витых пар подразделяются на неэкранированные и экранированные (рис. 3.2).

В свою очередь, среди экранированных конструкций различают кабели с общим внешним экраном, с экранами для каждой пары и с одновременным экранированием отдельных пар и сердечника в целом. Экранирование применяют для повышения переходного затухания (NEXT), снижения уровня ЭМИ и для повышения помехозащищенности.

Наибольшее распространение для экранирования отдельных пар получили металлизированные алюминием тонкие полимерные пленки, причем известны конструкции с ориентацией стороны металлизации как внутрь, так и наружу. Внешние экраны, окружающие кабельный сердечник, изготавливаются из такой же пленки или же выполняются в виде оплетки из оцинкованной медной проволоки. В состав конструкции пленочного экрана обычно вводится дополнительно тонкий неизолированный медный луженый или оцинкованный дренажный проводник диаметром около 0,5 мм.

В функции последнего входит обеспечение электрической непрерывности экрана при случайных разрывах пленки во время прокладки и эксплуатации.

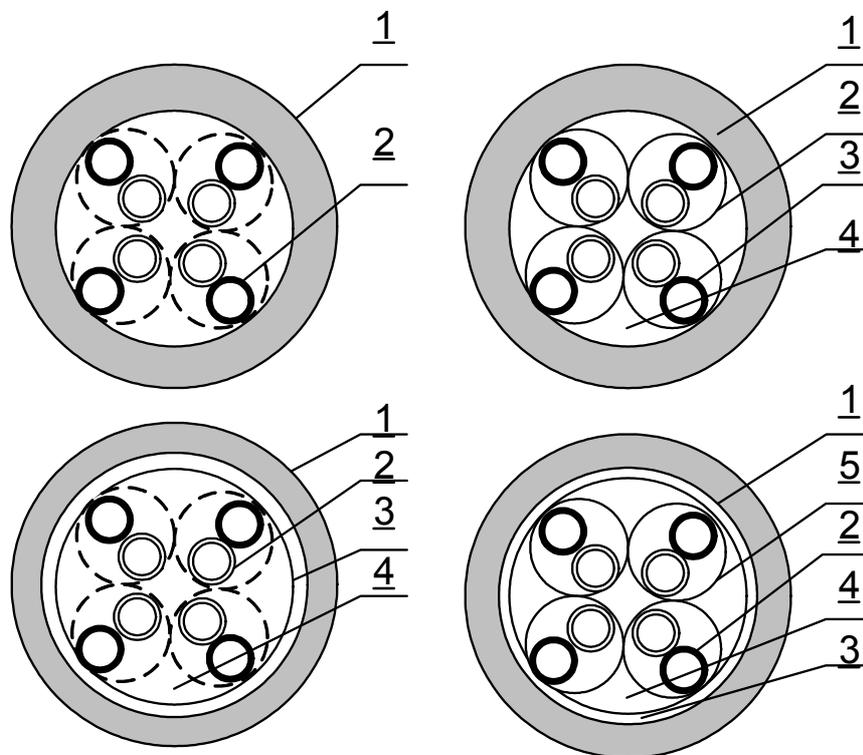


Рис. 3.2. Экранированный и неэкранированный кабель типа “витая пара”:

*1 – внешняя оболочка, 2 – витая пара; 3 – общий экран;
4 – дренажный провод; 5 – экран витой пары*

На практике получили достаточно широкое распространение кабели "витая пара" с общим пленочным экраном, который дополняется оплеткой. Пленочные экраны хорошо защищают кабель от высокочастотных помех (RFI), а экраны в виде оплетки – от низкочастотных (EMI), то есть двухслойный экран рассматриваемого вида обеспечивает надежное экранирование кабельного сердечника во всем диапазоне частот (табл. 3.2).

Экранирование кабелей

Условное обозначение		Экран	Цель экранирования
основное	альтернативное		
UTP	–	Отсутствует	–
STP	–	Экранирование каждой пары	Снижение уровня ЭМИ. Повышение защищенности от внешних помех
–	PiMF	Индивидуальный пленочный экран каждой пары	Повышение переходного затухания
S/UTP	STP, FTP	Общий экран для всех пар	Снижение уровня ЭМИ. Повышение защищенности от внешних помех
S/STP	STP, S-STP	Экранирование каждой пары плюс общий экран вокруг всех пар	Снижение уровня ЭМИ. Повышение защищенности от внешних помех. Повышение переходного затухания. Увеличение механической прочности

Областью применения кабелей S/UTP является построение горизонтальной подсистемы СКС при значительном уровне внешних наводок (производственные цеха и другие помещения с источниками сильных электромагнитных полей) или при повышенных требованиях к безопасности кабельной системы защиты от несанкционированного доступа.

S/STP-кабели обладают по сравнению с STP улучшенными характеристиками по защите от внешних помех и по уровню ЭМИ, однако основным их преимуществом перед другими конструктивными решениями является значительно более высокое (на 10...15 дБ и более при условии правильного монтажа) значение NEXT.

На сегодняшний день считается, что обеспечить передачу линейных сигналов с тактовой частотой свыше 250 – 300 МГц на требуемое стандартами расстояние 90 м можно только с использованием конструкции S/STP.

STP- и S-STP-кабели следует применять во всех случаях, перечисленных для S/UTP-кабелей, в тех ситуациях, когда:

- требуется получение кабельных сегментов, превышающих по длине 90 м;
- при построении систем передачи данных, для которых электрические характеристики кабелей категории 5 являются недостаточными;
- должны выполняться повышенные требования по защите от несанкционированного доступа к передаваемой информации.

Хотя параметры кабелей с индивидуальной экранировкой каждой пары могут существенно превосходить требования категории 5 (особенно по параметру NEXT и соответственно по параметру ACR), следует иметь в виду, что пока не существует стандартов ни на увеличенные длины сегментов, ни на сети, для работы которых электрические характеристики неэкранированных витых пар категории 5 являются недостаточными.

UTP-кабели в сравнении с экранированными обладают следующими преимуществами:

- меньшая стоимость;
- меньшая трудоемкость монтажа и эксплуатации;
- отсутствие повышенных требований к внутреннему заземляющему контуру здания;
- лучшие массогабаритные показатели;
- меньший радиус изгиба.

Основными преимуществами экранированных конструкций являются потенциально лучшая защита от внешних электромагнитных наводок, повышенная механическая прочность в случаях применения оплеточных экранов и более эффективная защита от несанкционированного доступа к передаваемой информации. Высокая теплопроводность экранов обеспечивает эффективный отвод тепла, которое возникает в проводниках в процессе передачи информации из-за протекания электрического тока.

На основании этого некоторые производители гарантируют для производимых ими экранированных конструкций меньшее затухание по сравнению с неэкранированными.

Сравнительная характеристика некоторых механических и эксплуатационных параметров основных вариантов конструкции четырехпарных горизонтальных кабелей приведена в табл. 3.3.

Таблица 3.3

Сравнительные характеристики четырехпарных горизонтальных кабелей

Тип кабеля	UTP		STP	S/UTP		S/STP
	Категория 5	Категория 6		Пленочный экран	Комбинированный экран	
Масса, кг/км	30 – 33	34 – 37	42	49	65 – 85	82 – 88
Внешний диаметр, мм	4,9	5,2	5,4	6,2	7,6	8,0
Рабочая температура, °С	–20, +60, +70					
Радиус изгиба, мм	30 – 35	–	–	35 – 40	40 – 45	–

Магистральный кабель, конструктивные особенности. Магистральный кабель предназначен для использования в магистральных подсистемах СКС для связи между собой помещений кроссовых.

В подсистеме внешних магистралей обычно большая часть маршрута прокладывается горизонтально, в подсистеме внутренних магистралей – вертикально. С целью снижения коэффициента затухания проводники изготавливаются из монолитной медной проволоки.

В отличие от горизонтального кабеля магистральные конструкции содержат более четырех витых пар и поэтому часто называются многопарными. Аналогично горизонтальным кабелям они различаются по категориям от 3 до 5, причем магистральные кабели категории 4 встречаются на практике очень редко. Конструкция кабеля зависит от емкости:

Категория кабеля	Количество пар
3	25, 50, 75, 100, 200, 300, 600, 900, 1800
5	25, 50, 100

При числе пар до 25 они помещаются в общую оболочку. При емкости свыше 25 пар они разбиваются на пучки по 25 пар в каждом, совокупность которых образует кабельный сердечник.

В некоторых конструкциях в качестве основы сердечника использован центральный стеклопластиковый стержень.

Провода одного пучка скрепляются полиэтиленовыми ленточками. Снаружи сердечник защищается общей диэлектрической оболочкой.

Кроме неэкранированных магистральных кабелей в ограниченном количестве производятся и S/UTP-конструкции, у которых под внешней диэлектрической оболочкой находится экран, закрывающий кабельный сердечник. Аналогично горизонтальным кабелям на их оболочку наносится маркировка, включающая в себя тип, данные по диаметру проводников и их количеству, наименование тестирующей лаборатории, а также футовые или метровые метки длины.

Погонная масса 25-парного кабеля категории 5 равна обычно 190 кг/км, рабочий диапазон температур составляет от -20 до $+60$ °С.

Кроме многопарных существуют так называемые многоэлементные (Multi Unit) кабели. Они отличаются тем, что кабельный сердечник образуют не отдельные витые пары, а двух- или четырехпарные элементы, аналогичные по конструкции горизонтальному кабелю и снабженные индивидуальной защитной оболочкой.

Для увеличения прочности и устойчивости к различным механическим воздействиям в качестве основы сердечника многоэлементного кабеля может применяться центральный стеклопластиковый прут.

Магистральные кабели подразделяются на кабели внутренней и внешней прокладки. Основным отличием кабеля внешней прокладки от внутриобъектного является применение специальных мер и конструктивных решений по защите кабельного сердечника от попадания в него влаги. Наиболее часто эта проблема решается использованием внешней полиэтиленовой оболочки.

Некоторые типы телефонных кабелей имеют гелиевое заполнение внутренних пустот сердечника.

Дополнительная защита кабельного сердечника от попадания влаги и механических воздействий выполняется броней из алюминиевой или стальной гофрированной ленты.

Горизонтальные кабели с граничной частотой свыше 100 МГц. В настоящее время на рынке компонентов СКС предлагается ряд типов серийных горизонтальных кабелей, характеристики которых существенно превышают требования стандартов к кабелям категории 5.

Общие черты неэкранированных конструкций:

- все они обеспечивают получение величины параметра ACR порядка 10 дБ на частотах примерно 150 – 200 МГц, то есть соответствуют характеристикам кабеля перспективной категории 6;

- увеличение параметра ACR достигнуто главным образом за счет улучшения параметра NEXT, хотя определенная доля может быть обеспечена уменьшением погонного затухания;

- характеристики кабелей нормируются до частот порядка 350 – 550 МГц из соображений использования их для передачи сигналов однонаправленных приложений, под которыми на практике в подавляющем большинстве случаев понимается многоканальное эфирное и кабельное телевидение.

При этом достаточно четко прослеживается деление рассматриваемых конструкций на два подкласса с граничными частотами соответственно 350 и 550 МГц.

Модели "младшего" подкласса часто отличаются от обычных кабелей категории 5 только несколькими лучшими значениями параметра NEXT и PS-NEXT, тогда как высокочастотные изделия имеют, наряду с улучшенными характеристиками переходного затухания, также меньшее затухание.

Дальнейшее увеличение рабочих частот горизонтальных кабелей обычных СКС без индивидуальной подборки параметров отдельных пар с возможностью их использования для сетей передачи данных (то есть по критерию $ACR = 10$ дБ), по мнению многих специалистов, при современном уровне техники возможно только на экранированных конструкциях.

Отметим, что многие кабельные заводы выпускают экранированные кабели из витых пар, характеристики которых нормируются на частотах вплоть до гигагерц.

Основной областью их применения считаются системы SOHO и передача сигналов приложений класса F. Для уменьшения затухания применяется увеличение диаметра медной жилы проводника до 0,55 мм против типовых для витой пары категории 5 значений 0,51 – 0,53 мм и использования изоляционных покрытий с уменьшенными диэлектрическими потерями, в частности, из вспененных материалов. Этим, кстати, объясняется несколько большие погонная масса и внешний диаметр по сравнению с кабелями категории 5.

Работы по увеличению параметра NEXT ведутся в двух направлениях. Первое из них основано на сохранении структуры сердечника в процессе прокладки и эксплуатации и базируется на введении в состав кабельного сердечника дополнительного элемента, выполняющего функции его силовой основы. В качестве такого элемента может быть использован центральный пластиковый пруток или полиэтиленовый профилированный элемент типа C (Central Crosstalk Cancellation) в форме четырехлучевой

звезды в поперечном сечении. Последний дополнительно за счет укладки каждой пары в индивидуальный паз разносит их друг от друга, что сопровождается заметным увеличением параметра NEXT.

Второе направление основано на поддержании высокой точности балансировки витых пар, то есть шага скрутки. Из соображений сохранения структуры сердечника во время прокладки поставка кабелей рассматриваемого вида выполняется в основном на катушках.

Структурные возвратные потери минимизируются ужесточением допусков на возможные флуктуации диаметра жилы, а также эксцентриситет жилы и изоляционной оболочки (до +1 % против типовых +3 % для конструкций категории 5).

Общая теория кабеля “витая пара”

Кабели данного типа зачастую сильно отличаются по качеству и возможностям передачи информации. Соответствия характеристик кабелей определенному классу или категории определяют общепризнанные стандарты (ISO 11801 и TIA-568). Сами характеристики напрямую зависят от структуры кабеля и применяемых в нем материалов, которые и определяют физические процессы, проходящие в кабеле при передаче сигнала. Если в кабеле присутствует более одной пары, то для исключения взаимных наводок пар, которые могли бы нарушить электромагнитный баланс, пары скручивают с различным шагом (рис. 3.3).

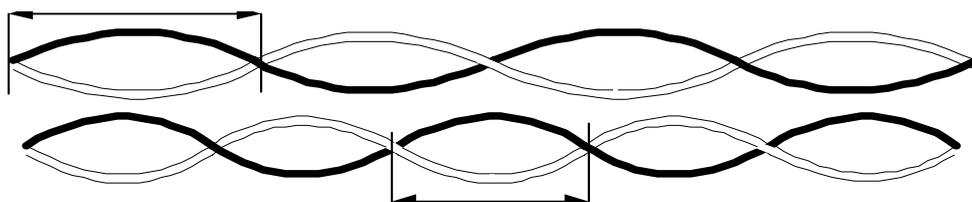


Рис. 3.3. Шаг скрутки проводов кабеля

Сбалансированность пары. Сбалансированность пары является фактически определяющей характеристикой качества кабеля, поскольку влияет на большинство других его свойств (рис. 3.4). Дело в том, что электромагнитное (Electro Magnetic – EM) поле наводит электрический ток в

проводниках и образуется вокруг проводника при протекании по нему электрического тока. Взаимодействие между ЭМ-полями и токонесущими проводниками может оказывать отрицательное воздействие на качество передачи сигнала. В обоих же проводниках сбалансированной пары электромагнитные помехи (em1 и em2) наводят одинаковые по амплитуде сигналы (S1 и S2), находящиеся в противофазе. За счет этого суммарное излучение "идеальной пары" стремится к нулю.

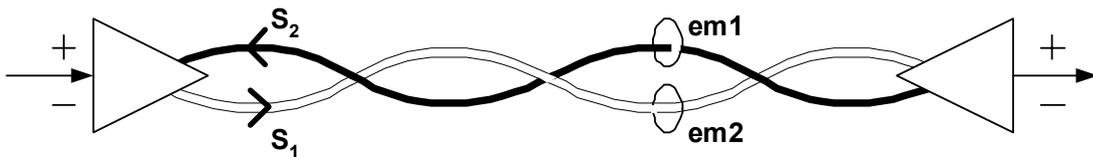


Рис. 3.4. Сбалансированная пара

Impedance (Характеристический импеданс). Как всякий проводник "витая пара" имеет сопротивление переменному электрическому току. Однако это сопротивление может быть различным для различных частот. "Витая пара" имеет импеданс обычно 100 или 120 Ом. В частности для кабеля категории 5 импеданс измеряется в диапазоне частот до 100 МГц и должен составлять $100 \text{ Ом} \pm 15 \%$ (рис. 3.5).

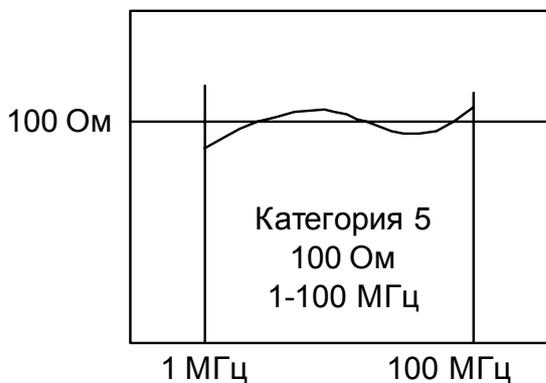


Рис. 3.5. Импеданс

Для идеальной пары импеданс должен быть одинаковым по всей длине кабеля, поскольку в местах неоднородности возникает эффект отражения сигнала, что, в свою очередь, может ухудшить качество

передачи информации. Чаще всего однородность импеданса нарушается при изменении в рамках одной пары шага скрутки, перегиба кабеля при прокладке или иного механического дефекта.

Скорость/задержка распространения сигнала. Скорость распространения сигнала NVP (Nominal Velocity of Propagation) выражается как отношение скорости распространения сигнала к скорости света.

Однако часто применяется производная от NVP и длины кабеля характеристика "delay" (задержка), выражающаяся в наносекундах на 100 м пары. Если в кабеле присутствует более одной пары, то вводят понятие "delay skew", или разность задержки. Дело в том, что пары не могут быть идеально одинаковы, что порождает разные задержки распространения сигнала в разных парах. Идеальные системы подразумевают, что подобные разницы будут минимальны.

Attenuation. Помимо импеданса и скорости распространения сигнала выделяют и другие важные характеристики кабеля типа "витая пара". Одной из них является погонное затухание (Attenuation), характеризующее величину потери мощности сигнала при передаче. Характеристика вычисляется как отношение мощности полученного на конце линии сигнала к мощности сигнала, поданного в линию. Поскольку величина затухания изменяется с ростом частоты, она должна измеряться для всего диапазона используемых частот. Сама величина выражается в децибелах на единицу длины.

На представленном графике (рис. 3.6) показаны потери мощности сигнала при передаче в зависимости как от длины кабеля, так и от используемой частоты.

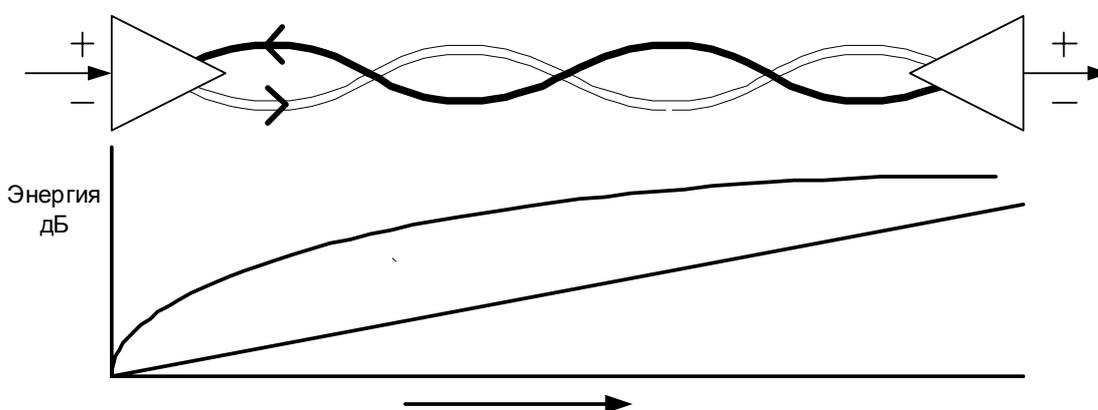


Рис. 3.6. Погонное затухание

NEXT (Near End Crosstalk). Другим важным параметром является переходное затухание между парами в многопарном кабеле, измеренное на ближнем конце, то есть со стороны передатчика сигнала, которое характеризует перекрестные наводки между парами. NEXT численно равен отношению подаваемого сигнала на одну пару к полученному наведенному в другой паре и выражается в децибелах. NEXT имеет тем большее значение, чем лучше сбалансирована пара.

Измерения необходимо проводить с обеих сторон, поскольку эта характеристика зависит от взаимного расположения измерительных приборов и мест возможных дефектов в кабеле. Как и погонное затухание, NEXT необходимо измерять для полного ряда частот.

В многопарном кабеле измерения производятся для всех комбинаций пар. Однако в настоящее время все чаще применяют и более глубокие тесты, основанные на выявлении групповых наводок на ближнем конце между всеми парами (Power Sum Crosstalk), присутствующими в кабеле.

Power Sum Crosstalk. Другое название данной характеристики – Power Sum NEXT, или PS-NEXT. Как и NEXT, Power Sum Crosstalk выражает переходное затухание между парами в многопарном кабеле, измеренное на ближнем конце, то есть со стороны передатчика сигнала. Однако учитываются одновременные наводки со всех пар, присутствующих в кабеле. Подобно NEXT, PS-NEXT измеряется с обоих концов линии для всего диапазона применяемых частот. Кроме оценки взаимных наводок пар на ближнем конце кабеля, переходное затухание измеряют и со стороны приемника сигнала. Данный тест получил название FEXT (Far End Crosstalk).

FEXT (Far End Crosstalk), или переходное затухание, на дальнем конце характеризует влияние сигнала в одной паре на другую пару. В отличие от NEXT FEXT измеряется посредством подачи тестового сигнала на пару в кабеле с одной пары и замера наведенного сигнала в другой паре со стороны приемника. Характеристика численно равна отношению тестового сигнала к наведенному посредством созданного электрического поля. FEXT как и все семейство характеристик переходного затухания, измеряется на всем диапазоне используемых частот и выражается в децибелах.

ACR (Attenuation Crosstalk Ratio). Одной из самых важных характеристик, отражающих качество кабеля является разность между погонным и переходным затуханиями, выражающемся в децибелах (рис. 3.7). Чем меньше погонное затухание, тем большую амплитуду имеет полезный сигнал на конце линии. С другой стороны, чем больше переходное затухание, тем меньше взаимные наводки пар. Таким образом, разность этих двух величин отображает реальную возможность выделения полезного сигнала

принимающим устройством на фоне помех. Для уверенного приема сигнала необходимо, чтобы Attenuation Crosstalk Ratio была не меньше заданного значения, определяемого стандартами для соответствующей категории кабеля. При равенстве погонного и переходного затуханий выделить полезный сигнал становится теоретически невозможно.

Так как характеристика не измеряется, а является результатом вычислений на основе измерений затухания, которые, в свою очередь, зависят от используемой частоты, ACR должна вычисляться для всего диапазона применяемых частот.

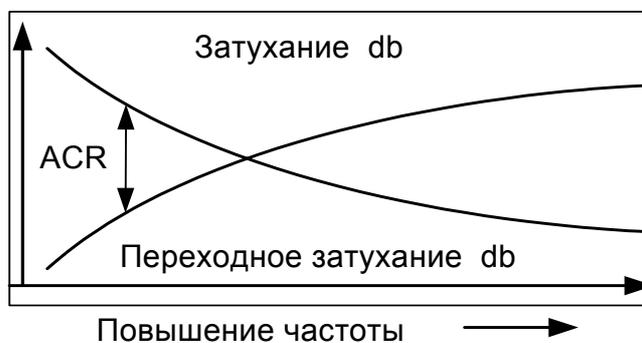


Рис.3.7. Затухание

ELFEXT (Equal Far End Crosstalk) – приведенное переходное затухание. Эта характеристика вычисляется на основании измерений переходного затухания на дальнем конце (FEXT) и погонного затухания (Attenuation) наводимой пары. Фактически ELFEXT – это ACR на дальнем конце кабельного линка, то есть разница между параметрами FEXT первой пары и Attenuation второй. ELFEXT, как и все семейство характеристик переходного затухания, вычисляется для всего диапазона используемых частот и выражается в децибелах.

PS-ELFEXT (Power Sum Equal Far End Crosstalk) – суммарное приведенное переходное затухание. Эта характеристика вычисляется для каждой отдельной пары простым суммированием значений ее параметров ELFEXT относительно всех остальных пар.

Return Loss (RL). При передаче сигнала возникает так называемый эффект отражения сигнала в обратном направлении или "обратное затухание". Величина отражения сигнала Return Loss, пропорциональна затуханию отраженного сигнала. Характеристика особенно важна при построении сетей с поддержкой протокола Gigabit Ethernet, использующего передачу сигналов по витой паре в обе стороны (полнодуплексная передача). Достаточно большой по амплитуде отраженный сигнал может исказить передачу информации в обратном направлении. Return Loss выражается в виде отношения мощности прямого сигнала к мощности отраженного.

Характеристики витых пар в зависимости от категории исполнения приведены в табл. 3.4.

Таблица 3.4

Электрические спецификации кабелей категории 3, 4 и 5

Параметр	Категория 3		Категория 4		Категория 5	
Число пар	4		4		4	
Импеданс	100 Ом ± 15 %		100 Ом ± 15 %		100 Ом ± 15 %	
Максимальное затухание (dB на 100 м, при 20 °С)	4 МГц	5,6	4 МГц	4,3	16 МГц	8,2
	10 МГц	9,8	10 МГц	7,2	31 МГц	11,7
	16 МГц	13,1	16 МГц	8,9	100 МГц	22
Переходное затухание (NEXT), дБ не менее	4 МГц	32	4 МГц	47	16 МГц	44
	10 МГц	26	10 МГц	41	31 МГц	39
	16 МГц	23	16 МГц	38	100 МГц	32

3.3. Оптический кабель

В основе оптоволоконных технологий лежит принцип использования света как основного источника информации. Отправитель преобразовывает информацию в световую волну, а адресат, получая последнюю, в свою очередь, интерпретирует свет как информацию.

Свет гораздо проще передать на дальние расстояния с меньшими потерями, нежели электрический ток. Кроме того, он не подвержен воздействию электромагнитных полей и способен передавать на порядки большее количество информации. С другой стороны, оптические технологии во многом являются более тонкими, поэтому качественная реализация оптоволоконного проекта требует детального понимания механизма передачи света и применяемых законов оптики.

Закон оптики. Породить световую волну довольно просто, не так-то просто ее сохранить и управлять ею. Однако это возможно, если использовать оптические законы распространения света. В оптоволоконных технологиях используется волновая теория света, то есть свет рассматривается как электромагнитная волна определенной длины. Для ее транспортировки используются изолированные оптически прозрачные среды. В однородной

среде электромагнитная волна распространяется прямолинейно, однако на границе изменения плотности среды ее направление и качественный состав меняются. В упрощенном варианте рассмотрим две граничащие среды с разной плотностью (рис. 3.8). Распространяясь в одной из них, луч может достигать поверхности другой под некоторым углом a (к нормали поверхности). При этом волна частично отражается в среду, из которой пришла, под углом b и частично проникает в новую среду в измененном направлении под углом c .

Согласно физическим законам распространения света угол падения луча равен углу отражения, то есть $a = b$. Также если обозначить величину плотности сред как n_1 и n_2 , то угол преломления c находится из соотношения $n_1 \cdot \sin a = n_2 \cdot \sin c$.

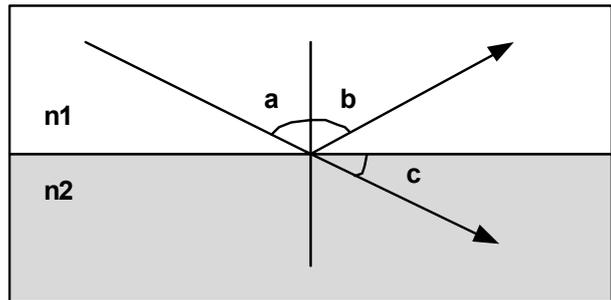


Рис. 3.8. Распространение сигнала

Эффект преломления света может отсутствовать, то есть возможна ситуация полного отражения света. Для этого достаточно, чтобы угол c был хотя бы нулевым. Трансформируя это выражение, получаем достаточное условие полного отражения света: $\sin a = n_2/n_1$. Именно за счет данного эффекта в современных оптоволоконных технологиях удается управлять распространением света в требуемой среде.

Принцип оптического волокна. Для того чтобы передать свет на большие расстояния, необходимо сохранить его мощность. Снизить потери при его передаче можно, во-первых, обеспечив достаточно оптически прозрачную среду распространения, тем самым сведя к минимуму поглощение волны, и, во-вторых, обеспечить правильную траекторию движения луча. Первая задача в настоящее время решается с помощью применения высокотехнологичных материалов, таких как чистое кварцевое стекло. Вторая задача решается с помощью закона оптики, описанного выше. За счет эффекта полного отражения света можно заставить луч "гулять" внутри ограниченной замкнутой среды, проделывая путь от источника сигнала до его приемника. Однако для этого необходимо две среды с разной плотностью.

Чаще всего в их качестве применяются кварцевые стекла различной плотности. Волну впускают в более плотную среду, ограниченную менее плотной. Среда вытягивают в так называемое оптическое волокно, сердцевину которого составляет более плотное стекло, в разрезе представляющее окружность и часто называемое световодом (рис. 3.9). Данный сердечник покрывают оболочкой из менее плотного стекла, при достижении которого транспортируемый сигнал будет полностью отражаться. Для предотвращения механических повреждений конструкция также снабжается защитной оболочкой, именуемой первичным покрытием.

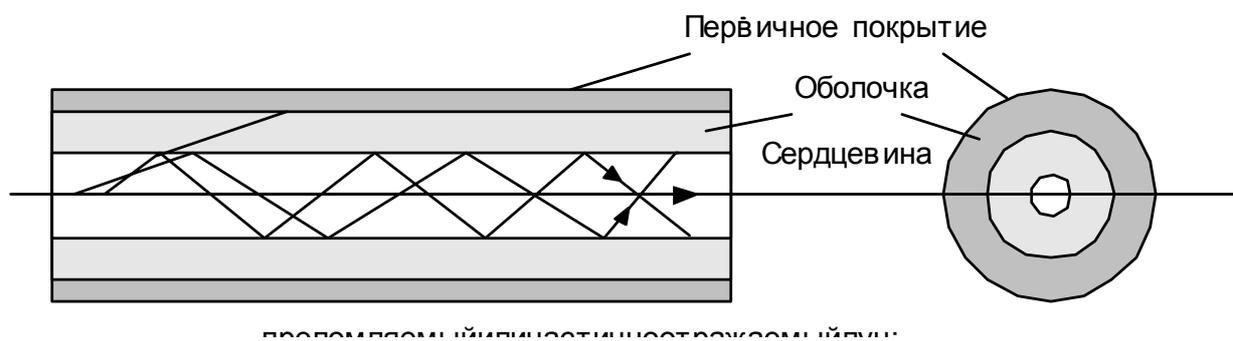


Рис. 3.9. Конструкция волокна;

- преломляемый или частично отражаемый луч;
- полностью отражаемый луч

Для достижения сигналом адресата необходимо впускать в сердцевину лучи под углом к боковой поверхности не менее критического. В этом случае реализуется эффект полного отражения, и теоретически луч никогда не покинет сердечника, кроме как через окончание волокна.

Однако на практике все же существует некоторый процент преломляемых лучей. Это связано, во-первых, со сложностью реализации подобного источника света, во-вторых, с невозможностью изготовления идеально ровного волокна, и в-третьих, с неидеальной инсталляцией оптического кабеля.

Межмодовая дисперсия. Поскольку источники излучения не идеальны, испускаемые ими волны не совсем идентичны и могут различаться

по направлению распространения. Единичная независимая траектория распространения волны именуется модой. Очевидно, что луч, направленный параллельно оси световода проходит меньшее расстояние, нежели луч, распространяющийся по траектории ломаной за счет эффекта отражения. Как следствие лучи достигнут конца сердечника в разные моменты времени.

При учете неидеальных свойств применяемых источников светового сигнала возможна ситуация, когда изначальный световой импульс содержит некоторое множество

волн, входящих в световод под разными углами. В итоге импульс раскладывается на множество отдельных волн, достигающих приемник в разные моменты. Именно этот разброс времени и называется межмодовой дисперсией (рис. 3.10).

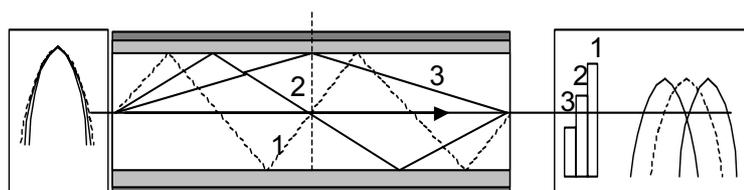


Рис. 3.10. Межмодовая дисперсия

Межчастотная дисперсия. Погрешность источников излучения еще состоит и в некотором разбросе генерируемых частот. Испускаемые волны не совсем идентичны и могут различаться по длине. Согласно законам физики более короткие волны распространяются быстрее, а следовательно, волны достигают конца световода в разные моменты времени (рис. 3.11).

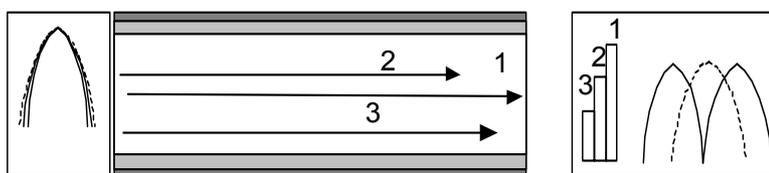


Рис. 3.11. Межчастотная дисперсия

При учете неидеальных свойств применяемых источников светового сигнала возможна ситуация, когда изначальный световой импульс содержит некоторое множество входящих в световод волн с разной частотой. В итоге импульс раскладывается на множество отдельных волн, достигающих приемник в разное время. Именно этот разброс времени и называется межчастотной дисперсией.

Материальная дисперсия. Скорость преодоления расстояний волной зависит не только от частоты, но и от плотности среды распространения. В применяемых в настоящее время световодах распределение плотно-

сти сердечника может быть неравномерным, как в случае с градиентными волокнами (об этом позже). Вследствие этого волны, проходящие путь по разным траекториям, обладают разными скоростями распространения и оказываются в приемнике в разное время (рис. 3.12).

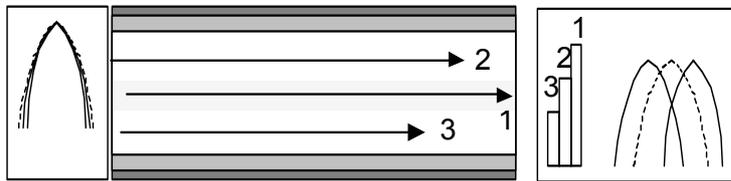


Рис. 3.12. Материальная дисперсия

При учете неидеальных свойств применяемых источников светового сигнала возможна ситуация, когда изначальный световой импульс содержит не-

которое множество волн, проходящих световод по разным траекториям, каждая из которых пересекает участки среды с разными плотностями. В итоге импульс раскладывается на множество отдельных волн, достигающих приемник в разное время. Именно этот разброс времени и называется материальной дисперсией.

Влияние дисперсии на пропускную способность канала. Дисперсия, будь то материальная, межчастотная или межмодовая, отрицательно влияет на пропускную способность канала. Дело в том, что современные оптоволоконные технологии используют цифровой способ передачи информации. Световой сигнал поступает импульсами. Чем сильнее размыт по времени импульс на выходе (эффект дисперсии), тем большие требуются интервалы между передаваемыми сигналами, что и ограничивает, в свою очередь, пропускную способность канала. Поэтому необходимо снижать величины дисперсий, тем самым увеличивая возможное количество информационных сигналов за единицу времени. Вообще из-за эффекта дисперсии необходимо пытаться сократить количество проникающих одновременно мод (лучей) в световод.

Многомодовое ступенчатое волокно. Основное различие между вариантами оптического волокна состоит в свойствах применяемого в них сердечника. Самый простой вариант сердечника – это кварцевое стекло с равномерной плотностью. Если отобразить плотности распределения слоев волокна, то получится ступенчатая картина, что и отображено в названии этого типа волокна (рис. 3.13). При достаточно большом радиусе равномерно плотного световода наблюдается эффект межмодовой дисперсии. Ее влияние на производительность оптического канала оказывается много

больше межчастотной и материальной. Поэтому при расчете пропускной способности канала пользуются именно ее показателями.

В настоящее время используют три стандартных диаметра сердечника многомодового волокна: 100, 62,5 и 50 микрон. Наиболее распространенны световоды диаметром

62,5 микрон, однако постепенно все более прочные позиции завоевывает сердечник диаметром 50 микрон. Вследствие простых

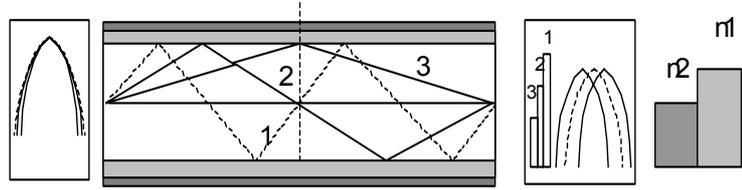


Рис. 3.13. Многомодовое ступенчатое волокно

геометрических законов распространения света несложно убедиться в его большой пропускной способности, поскольку он пропускает меньшее количество мод, тем самым уменьшая дисперсию импульса на выходе. Размер световодов выбран не случайно. Он непосредственно связан с используемой частотой световой волны. На данный момент выделяют три основные длины волны: 850, 1300 и 1500 нм. Почему выбраны именно эти длины волн, мы поясним позже.

Многомодовые ступенчатые волокна обладают малой пропускной способностью относительно действительных возможностей света, в связи с этим чаще в многомодовой технологии используют градиентные волокна.

Многомодовое градиентное волокно. Название волокна говорит само за себя. Основное отличие градиентного волокна от ступенчатого заключается в неравномерной плотности материала световода. Если отобразить плотности распределения на графике, то получится параболическая картина. Эффект межмодовой дисперсии, как и в случае ступенчатой схемы, все же проявляется, однако намного меньше. Это легко объяснимо с точки зрения геометрии. На рис. 3.14 видно, что длины пути лучей сильно сокращены за счет сглаживания. Более того, интересен тот факт, что лучи, проходящие дальше от оси световода, хотя и преодолевают большие расстояния, но при этом имеют большие скорости, так как плотность материала от центра к внешнему радиусу уменьшается. А световая волна распространяется тем быстрее, чем меньше плотность среды.

В итоге более длинные траектории компенсируются большей скоростью. При удачно сбалансированном распределении плотности стекла

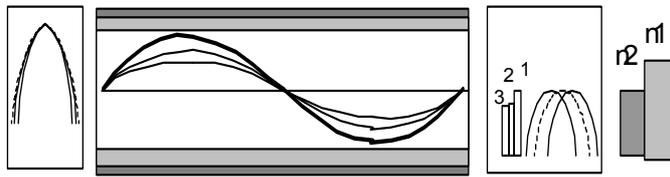


Рис. 3.14. Многомодовое градиентное волокно

можно свести к минимуму разницу во времени распространения, за счет этого межмодовая дисперсия градиентного волокна уменьшается. Как и в случае со ступенчатым волокном, в на-

стоящее время используют три стандартных диаметра градиентного сердечника: 100, 62,5 и 50 микрон, работающих также на частотах 850, 1300 и 1500 нм. Однако насколько не были бы сбалансированы градиентные многомодовые волокна, их пропускная способность не сравнится с одномодовыми технологиями.

Одномодовое волокно. Согласно законам физики при достаточно малом диаметре волокна и соответствующей длине волны через световод будет распространяться единственный луч. Вообще сам факт подбора диаметра сердечника под одномодовый режим распространения сигнала говорит о частности каждого отдельного варианта конструкции световода. То есть при употреблении понятий «много-» и «одномодовый» следует понимать характеристики волокна относительно конкретной частоты используемой волны (рис. 3.15).

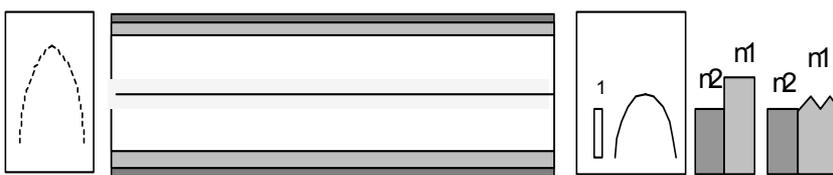


Рис. 3.15. Одномодовое ступенчатое градиентное волокно

Распространение лишь одного луча позволяет избавиться от межмодовой дисперсии. Как уже отмечалось, именно эта дисперсия имеет наибольшее влияние

на пропускную способность канала. Величины материальной и межчастотной дисперсии на порядок меньше межмодовой. Однако одномодовое волокно исключает возможность распространения нескольких лучей, поэтому межмодовая дисперсия отсутствует, в связи с чем одномодовые световоды на порядок производительнее. На данный момент применяется сер-

дечник с внешним диаметром около 8 микрон. Как и в случае с многомодовыми световодами, используется и ступенчатая, и градиентная плотность распределения материала. Второй вариант более производительный. Одномодовая технология более тонкая, дорогая и применяется в настоящее время в телекоммуникациях, многомодовые же кабели завоевали свою нишу в локальных компьютерных сетях.

Затухание сигнала, окна прозрачности. Кроме сложностей, связанных с уменьшением дисперсии волны, существует и проблема сохранения мощности передаваемого сигнала. Хотя световую волну сохранить легче, чем электрический ток, она испытывает эффекты поглощения и рассеивания. Первый связан с преобразованием одного вида энергии в другой. Так, волна определенной длины порождает в некоторых химических элементах изменение орбит электронов, в других – происходит резонанс. Это, в свою очередь, и порождает преобразование энергии. Известно, что поглощение волны тем меньше, чем меньше ее длина. В связи с этим применять чрезмерно длинные волны невозможно, так как резко возрастают потери при нагреве световодов. Однако, с другой стороны, безгранично снижать длины волн тоже нецелесообразно, так как в этом случае возрастают потери на рассеивание сигнала. Именно баланс рассеивания и поглощения волны определяет диапазон применяемых волн в оптоволоконных технологиях (рис. 3.16).

Теоретически лучшие показатели достигаются на пересечении кривых поглощения и рассеивания. На практике зависимость затухания несколько сложнее и связана с химическим составом среды, в которой распространяется волна. В световодах основными химическими элементами являются кремний и кислород, каждый из которых проявляет активность на определенной частоте волны, с чем связано ухудшение теоретической прозрачности материала световода в двух окрестностях. В итоге образуются три окна в диапазоне длин волн. В рамках этих окон

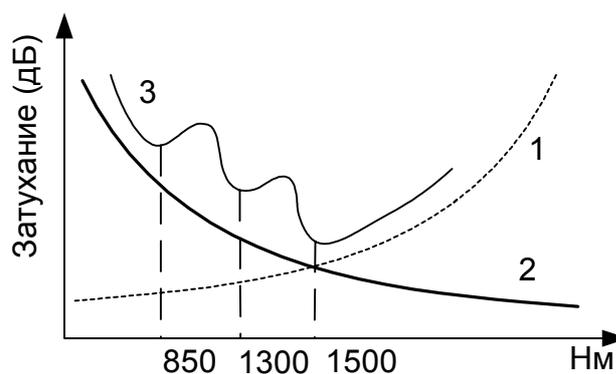


Рис. 3.16. Потери мощности в световоде:
1 – за счет поглощения волны; 2 – за счет рассеивания волны; 3 – практические потери с учетом химического состава среды распространения волны

В итоге образуются три окна в диапазоне длин волн. В рамках этих окон

затухание волны имеет наименьшее значение. Сам параметр оптических потерь измеряется в децибелах на километр.

Используемые длины волн. Именно "окна прозрачности" определили длины волн, которые применяются в современных оптоволоконных технологиях. Чаще всего это три длины: 850, 1300 и 1500 нм. Наиболее качественной и высокоскоростной связью обладают каналы на основе волн длиной 1500 нм. Однако окончное оборудование, способное работать на данной длине волны, значительно дороже и предполагает применение только лазерных источников света. Поэтому зачастую возникает проблема оценки экономической целесообразности применения подобных сетей.

Рабочая длина волны 850 нм наиболее характерна для многомодовых волокон, тогда как одномодовые волокна применяются для волн длиной на 1500 нм.

Теория оптических коннекторов. Очевидно, что в идеальной оптической системе передачи информации световой поток должен беспрепятственно проходить трассу от источника до фотоприемника. Оптическое волокно – это нечто иное, как та же самая трасса распространения сигнала. Протянуть цельное волокно от источника до приемника не представляется возможным. Технологическая длина волокна обычно не превышает нескольких километров. И если эту проблему еще можно решить сваркой световодов, то обеспечение мобильности локальной оптической подсети достигается только с применением кроссового оборудования. Проблем передачи световой волны от одного отрезка волокна к другому не избежать. Для многократного и простого подключения оптических линков световоды могут оконцовываться оптическими коннекторами. Учитывая, что современные световоды – это микронные технологии, оконцовка волокна оптическими коннекторами представляет собой непростую задачу.

Потери в оптических коннекторах. Опишем проблемы, возникающие при переходе сигнала из одного световода в другой. Потеря мощности или затухание оптической волны возникает при неточной центровке световодов (рис. 3.17). В этом случае часть лучей просто не переходит в следующий световод или входит под углом более критического.

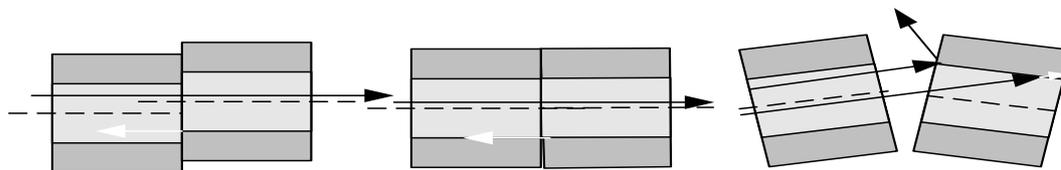


Рис. 3.17. Смещение оси световода

При неполном физическом контакте волокон образуется воздушный зазор. В связи с чем возникает эффект возвратных потерь. Часть лучей при прохождении прозрачных сред с разной плотностью отражается в обратном направлении. Достигая резонатора, они усиливаются и вызывают искажения сигналов.

Неидеальная геометрическая форма волокон также вносит вклад в потери мощности. Это может быть и эллиптичность световода и смещение оси относительно его сердцевины. Торец самого световода может содержать деформации: сколы и шероховатости, что, в свою очередь, уменьшает рабочую поверхность соприкосновения волокон (рис. 3.18).

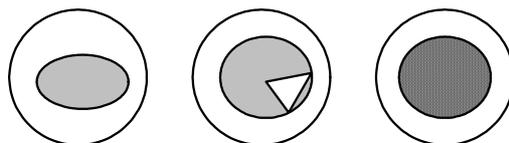


Рис. 3.18. Форма торцов

Наконечники оптических коннекторов. Необходимо точно и плотно совместить оба световода. Чтобы обеспечить сохранность хрупкого волокна при многократном совмещении, их оконечные отрезки помещают в керамические, пластмассовые или стальные наконечники. Большинство наконечников имеют цилиндрическую форму с диаметром 2,5 мм. Встречаются конические конструкции, а коннекторы LC имеют наконечник диаметром 1,25 мм.

Внутри наконечников существует канал, в который вводится и фиксируется химическим или механическим способом очищенный от оболочки световод. При удалении защитного покрытия могут использоваться как специальные механические инструменты, так и химически активные растворы. Внутри наконечника световод может фиксироваться как по всей длине канала (чаще это методы на основе клея), так и в точке ввода волокна в наконечник (механические методы). Процесс механической фиксации занимает гораздо меньше времени (до нескольких минут) и основан на "придавливании" волокна с помощью полимерных материалов. Но он является менее надежным и недолговечным. Химический способ говорит сам за себя. Чаще всего фиксирующим составом в данной технологии выступают эпоксидные растворы как наиболее надежные. Однако период полного загустевания такого состава весьма продолжителен – до суток. Поэтому

при необходимости более быстрого монтажа коннекторов могут применяться другие компоненты или специальные печи для сушки.

После установки световода в коннектор необходимо отшлифовать торец наконечника. Выступающий излишек волокна удаляется специальными инструментами. Основным принцип заключается в надрезе и обламывании световода, после чего можно приступить к непосредственной полировке поверхности.

Особый интерес вызывает форма торцов наконечников. Их обработка представляет собой целое искусство. Простейший вариант торца – плоская форма. Ей присущи большие возвратные потери, поскольку вероятность возникновения воздушного зазора в окрестности световодов велика. Достаточно неровностей даже в нерабочей части поверхности торца. Поэтому чаще применяются выпуклые торцы (радиус скругления составляет порядка 10 – 15 мм). При хорошем центрировании плотное соприкосновение световодов гарантируется, а значит более вероятно отсутствие воздушного зазора. Еще более продвинутым решением является применение скругления торца под углом в несколько градусов. Скругленные торцы меньше зависят от деформаций, образуемых при соединении коннекторов, поэтому подобные наконечники выдерживают большое количество подключений (от 100 до 1000).

Также важен материал наконечника. Подавляющее число коннекторов строятся на основе керамических наконечников как более стойких. После оконцовки световодов коннекторами необходимо произвести анализ качества поверхности наконечника. Чаще всего для этого применяются микроскопы. Профессиональные приборы обладают кратностью увеличения в сотни раз и снабжены специальной подсветкой с различных ракурсов. Они могут также иметь интерфейс подключения к дополнительному измерительному оборудованию.

Согласно стандарту TIA/EIA 568A величина возвратных потерь для многомодового волокна в оптических коннекторах не должна превышать 20 дБ, а для одномодового – 26 дБ. По величине возвратных потерь коннекторы делятся на классы (PC – от англ. Physical Contact):

Тип	Потери
PC	< 30 дБ
Super PC	< 40 дБ
Ultra PC	< 50 дБ
Angled PC	< 60 дБ

Соединение оптических коннекторов

Принципиально соединение двух оптических коннекторов кроссового оборудования строится по следующей схеме: платформой для установки коннекторов служит розетка; входящие в нее коннекторы фиксируются таким образом, чтобы оси их наконечников (керамика) были отцентрированы, параллельны и плотно прижаты. Подобные розетки обычно устанавливаются в патч-панели или вставке монтажных коробов.

Потери в наконечниках соединителей для основных типов коннекторов в зависимости от вида физической среды приведены в табл. 3.5.

Таблица 3.5

Уровни потерь в коннекторах

Тип коннектора	Потери (дБ) при 1300 нм	
	Многомодовый	Одномодовый
ST	0,25	0,3
SC	0,2	0,25
LC	0,1	0,1
FC	0,2	0,6
FDDI	0,3	0,4

ST-коннектор (рис. 3.19). Коннекторы различаются не только применяемыми наконечниками, но и типом фиксации конструкции в розетке. Самым распространенным представителем в локальных оптических сетях является ST-тип коннектора (от англ. Straight Tip). Керамический наконечник имеет цилиндрическую форму диаметром 2,5 мм со скругленным торцом. Фиксация производится за счет поворота оправы вокруг оси коннектора, при этом вращения основы коннектора отсутствуют (теоретически) за счет паза в разъеме розетки.

Направляющие оправы, сцепляясь с упорами ST-розетки, при вращении вдавливают конструкцию в гнездо. Пружинный элемент обеспечивает необходимое прижатие.



Рис. 3.19. ST-коннектор

Слабым местом ST-технологии является вращательное движение оправы при подключении/отключении коннектора. Оно требует большого

жизненного пространства для одного линка, что важно в многопортовых кабельных системах. Более того, вращения наконечника отсутствуют только теоретически. Даже минимальные изменения положения последнего влекут рост потерь в оптических соединениях. Наконечник выступает из основы конструкции на 5 – 7 мм, что ведет к его загрязнению.

SC-коннектор (рис. 3.20). Слабые стороны ST-коннекторов в настоящее время решаются за счет применения SC-технологии (от англ. Subscriber Connector). Сечение корпуса имеет прямоугольную форму. Подключение/отключение коннектора осуществляется поступательным движением по направляющим и фиксируется защелками. Керамический наконечник также имеет цилиндрическую форму диаметром 2,5 мм со скругленным торцом (некоторые модели имеют скос поверхности). Наконечник почти полностью покрывается корпусом и потому менее подвержен загрязнению, нежели в ST-конструкции. Отсутствие вращательных движений обуславливает более осторожное прижатие наконечников.

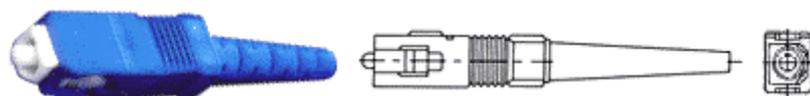


Рис. 3.20. SC-коннектор

В некоторых случаях SC-коннекторы применяются в дуплексном варианте. На конструкции могут быть предусмотрены фиксаторы для спаривания коннекторов, или используются специальные скобы для группировки корпусов. Коннекторы с одномодовым волокном обычно имеют голубой цвет, а с многомодовым – серый.

LC-коннектор (рис. 3.21). Коннекторы типа LC – это малогабаритный вариант SC-коннекторов. Он также имеет прямоугольное сечение корпуса. Конструкция выполняется на пластмассовой основе и снабжена защелкой, подобной защелке, применяющейся в модульных коннекторах медных кабельных систем. Вследствие этого и подключение коннектора производится схожим образом. Наконечник изготавливается из керамики и имеет диаметр 1,25 мм.

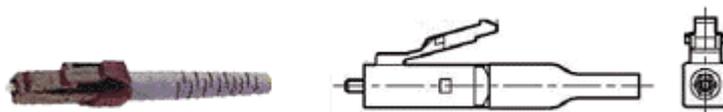


Рис. 3.21. LC-коннектор

Встречаются как многомодовые, так и одномодовые варианты коннекторов. Ниша этих изделий – многопортовые оптические системы.

FC-коннектор (рис. 3.22). В одномодовых системах встречается еще одна разновидность коннекторов – FC. Он характеризуется отличными геометрическими характеристиками и высокой защитой наконечника.



Рис. 3.22. FC-коннектор

FDDI-коннектор (рис. 3.23). Для подключения дуплексного кабеля могут использоваться не только спаренные SC-коннекторы. Часто в этих целях применяют FDDI-коннекторы. Конструкция выполняется из пластмассы и содержит два керамических наконечника. Для исключения неправильного подключения линка коннектор имеет несимметричный профиль.

Технология FDDI предусматривает четыре типа используемых портов: А, В, S и М. Проблема идентификации соответствующих линков решается за счет снабжения коннекторов специальными вставками, которые могут различаться по цветовой гамме или содержать буквенные индексы.

В основном данный тип используется для подключения к оптическим сетям оконечного оборудования.



Рис. 3.23. FDDI-коннектор

MT-RJ-коннекторы. Разработчики коннектора MT-RJ преследовали решение следующих задач: малый размер, низкая стоимость и простота установки. Использование коннектора MT-RJ увеличивает плотность портов в два раза по сравнению со стандартными коннекторами и делает его идеальным для использования в приложениях типа fiber to the desk. Дизайн коннектора соответствует требованиям TIA. Размер и конструкция защелки аналогичны модели RJ-45 :

- Дуплексный ферул.
- Низкая стоимость.
- Высокая плотность портов.

- Соответствие стандартам ISO/IEC 11801 и TIA/EIA 568A .
- Низкие прямые потери: < 0,22 дБ для ММ, < 0,19 дБ для ОМ.

В коннекторе MT-RJ используется улучшенная версия индустриального стандарта для коннекторов типа RJ-45. Именно малый размер и удобство защелки, аналогичной модели RJ-45, определяют преимущества данного коннектора при использовании в горизонтальной проводке до рабочего места.

Отличием системы MT-RJ от Molex является использование различных РН для коннекторов модификаций «вилка» (с направляющими штырьками, выступающими из ферула) и «розетка» (с дырочками под штырьки). Имеются две модификации адаптера, одна из которых устанавливается в гнездо для симплексного SC-адаптера.

Оптимальное подключение волоконно-оптических кабелей

Компьютерные сети крупных и средних организаций довольно часто оказываются перегружены довольно значительными потоками данных, возникающих в процессе работы с графикой, различных операций с базами данных большого объема, цифрового моделирования и управления процессами в реальном времени и выполнения многих других задач. Причем прослеживается тенденция постоянного увеличения объема этих потоков.

Решением данной проблемы может оказаться замена кабельной проводки на более высокопроизводительную, так как реализация более скоростных сетевых технологий на существующей проводке может не дать существенного выигрыша (табл. 3.6). Предпочтительно в качестве альтернативы существующим использовать волоконно-оптические кабели, тем более что имеются реализации перспективных сетевых технологий, например гигабитного Ethernet или АТМ на 622 Мбит/с, которые предполагают их применение.

Таблица 3.6

Качество и характеристики волоконного кабеля

Характеристики	ОМ-кабель	ММ-кабель
Диаметр волокна, мкм	125	125
Прямые потери (среднее значение + дисперсия), дБ	0,19; 0,06	0,22; 0,09
Минимальные обратные потери, дБ	45	20

<i>Окончание табл. 3.6</i>		
Характеристики	ОМ-кабель	ММ-кабель
Гарантированная прочность закрепления кабеля, Н	67	67
Изменение прямых потерь на коннекторе после 200 включений, дБ	< 0,03	< 0,05
Температурная устойчивость (– 40 °С... +75 °С) изменения прямых потерь, дБ	< 0,15	< 0,15
Материал ферула	Термопластик	Пластик

Оконцовка и соединение волоконно-оптических кабелей представляют собой довольно сложную техническую задачу, оптимальное решение которой определяет эффективность всего проекта в целом. Особенно это ощутимо при соединении группы зданий, поскольку свободные манипуляции с магистральным волоконно-оптическим кабелем затруднены и согласно техническим требованиям недопустимы, для дальнейшей разводки после ввода необходимо осуществить подключение кабеля к соединительному оборудованию. Для его подключения могут использоваться так называемые пигтейлы (pigtail – буквально свиной хвост, который представляет собой отрезок оптического световода, на одном конце которого устанавливается коннектор, а другой остается свободным). Этот свободный конец соединяется с кабелем путем реализации неразъемного соединения, преимущественно сварного. Для его осуществления требуется произвести разделку и зачистку кабеля, далее заделку кабеля и волокон (поскольку преимущественно приходится иметь дело с кабелями со свободным буфером), подготовку торцов и контроль их качества, состыковку и, непосредственно, сварку с последующим контролем качества шва и оптических характеристик соединения, и укладку соединений в специальные кассеты или муфты, чтобы предохранить их от повреждений. Ко всему этому следует добавить вполне возможное несоответствие полученных параметров норме или просто ошибки, что влечет за собой повторение процедуры.

Все перечисленные операции требуют привлечения специально подготовленного персонала, обучение которого – дело довольно дорогостоящее, а также специального оборудования и инструмента, стоимость которых также весьма существенна, что и определяет высокий уровень затрат на прокладку и подключение кабеля.

Альтернативой может служить использование кабелей с установленными производителем коннекторами. Но такое решение не вполне пригодно для прокладки кабеля в кабельных каналах, поскольку сами по себе коннекторы при сравнении с оптическими волокнами являются довольно объемными. Эти коннекторы заключаются в специальные тяговые протекторы, то есть на конце кабеля имеется довольно объемистая конструкция, диаметр которой во много раз превосходит диаметр кабеля. Для протяжки такого кабеля потребуются, соответственно, большее пространство и большие усилия.

Принцип применения технологии MT. Некоторые компании, в частности BICC Brand Rex, предлагают упрощенную технологию подключения магистральных волоконно-оптических кабелей. Она по праву может считаться технологией plug-and-play и представляет готовое решение указанной проблемы для небольших проектов при соединении нескольких зданий и реализации вертикальной разводки. Основу ее составляет применение устанавливаемых производителем на соответствующие кабели MT или MTP коннекторов, посредством которых осуществляется соединение 4, 8 или 12 оптических волокон, и коммутационных панелей (так называемых патч-панелей или кросс-панелей), оснащенных веерообразной сборкой соединительных кабелей (fanout unit) со вторым коннектором (рис. 3.24). На концах соединительных кабелей устанавливаются стандартные ST-, FC- или SC-коннекторы.

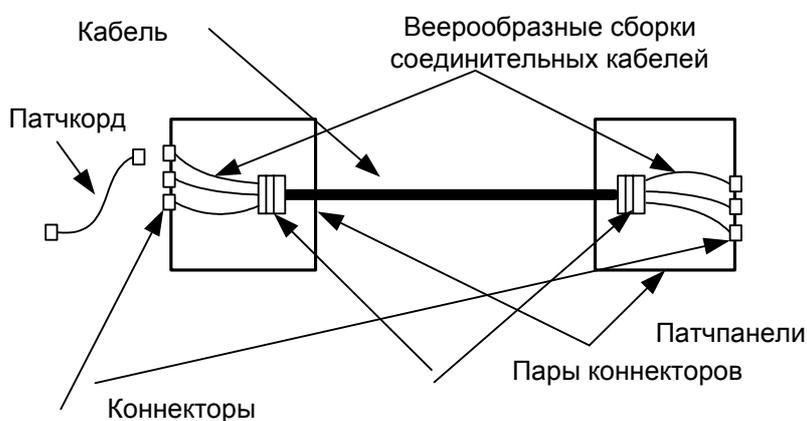


Рис. 3.24. Прокладка и соединение кабеля

Прокладка и подключение волоконно-оптических кабелей с *MT*- или *MTP*-коннекторами, установленными производителем, не требуют применения специального инструмента и привлечения квалифицированного персонала, поскольку нет необходимости производить оконцовку кабеля. При этом обеспечиваются высокие характеристики соединения. Возможна про-

кладка в помещениях и вне помещений при соединении нескольких зданий магистральным кабелем и реализации вертикальной разводки.

Опыт применения этой технологии во многих странах многократно подтверждает высокую надежность и другие преимущества ее использования, за счет чего обеспечивается недорогой, быстрый и эффективный монтаж волоконно-оптических кабелей.

MT/MTP-коннектор (рис. 3.25). Коннекторы MT разработаны телекоммуникационными компаниями коннекторов, которые представляют революционную технологию в соединении волоконно-оптических кабелей. В MT-коннекторе осуществляется совмещение полосок, содержащих 4, 8 или 12 оптических волокон. Эти коннекторы отличает высокая надежность, а характеристики передачи не хуже, чем у обычных коннекторов для волоконно-оптических кабелей. Оптические характеристики соединения MT-коннектором приведены в табл. 3.9.

Для соединения MT-коннекторов между собой используются два установочных направляющих штырька и пружинный фиксатор.

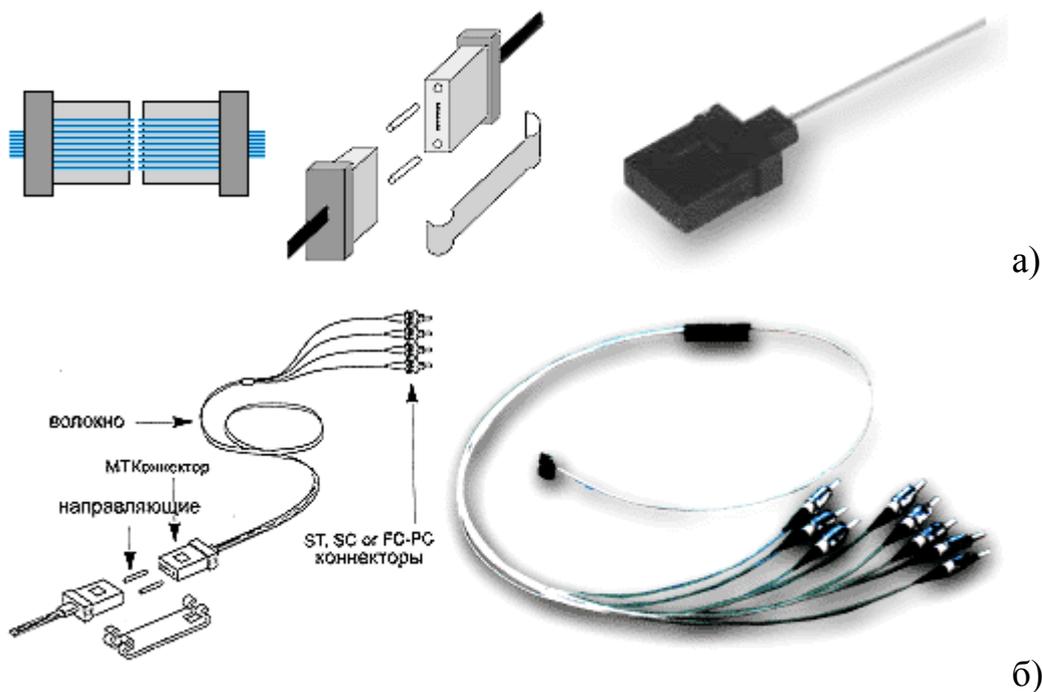


Рис. 3.25. MT/MTP-коннектор. Конструктивное оформление разъемов (а) и соединителей (б)

Соединительные кабели, которые составляют веерообразную сборку, – это короткие отрезки оптического волокна, один конец которых устанавливается в MT-коннектор, а на другом – обычный коннектор типа ST, SC или FC-PC. Таких отрезков может быть 4 или 8, или 12. Применяется во-

локно типов 50/125, 62,5/125 и одномодовое. Веерообразная сборка соединительных кабелей монтируется в универсальную патч-панель. МТ-коннектор кабеля присоединяется к МТ коннектору сборки, а обычные коннекторы подключаются к соответствующим адаптерам на патч-панели.

Таким образом, можно соединять до 24 волокон в одной патч-панели (при этом используются два комплекта МТ-коннекторов). В оконцованных производителем кабелях могут содержаться по 4, 8 или 12 волокон типа 50/125, 60/125 или одномодовое. Кабели могут быть разнообразной конструкции и иметь наружные оболочки любого типа. Стандартная максимальная длина 500 м. Могут изготавливаться кабели заказной длины (до 4 км, рекомендуется не больше 1 км).

Для защиты коннекторов при прокладке в кабельных каналах на концы кабеля надеваются тяговые протекторы. Кабель наматывается на катушки с тремя фланцами. За счет особенностей намотки и конструкции катушки обеспечивается доступ к обоим концам кабеля.

Реализация описанной технологии содержит две патч-панели. Они оснащены веерообразными сборками соединительных кабелей и кабелем требуемой длины с установленными на концах МТ- и МТР-коннекторами. Все что требуется на начальном этапе – это несколько раз тщательно измерить расстояние между предполагаемыми точками подключения кабеля и к полученной величине добавить некоторое значение, которое учитывает припуски и укладку кабеля. Следует учитывать, что промежуточные соединения для данной технологии не допускаются. После прокладки кабеля достаточно подсоединить его к патч-панелям. Таким образом, подключение волоконно-оптического кабеля с использованием предложенной технологии доступно всем.

Таблица 3.7

Потери в световодах

Тип световода	Вносимые потери		Возвратные потери	
	Норма	Максимум	Норма	Минимум
Одномодовый, плоская полировка, заполненный иммерсионным гелем	0,35	1,0	45	35
Одномодовый, плоская полировка	0,5	1,3	25	15
Одномодовый, полировка под углом	0,8	1,5	60	50
Многомодовый, плоская полировка	0,3	0,8	–	–

3.4. Коаксиальный кабель

Представляет собой электрический кабель, состоящий из центрального провода 1 и металлической оплетки 2, разделенных между собой слоем диэлектрика 3 (внутренней изоляции) и помещенных в общую внешнюю оболочку 4 (рис. 3.26). Коаксиальный кабель до недавнего времени был распространен очень широко, что связано с его высокой помехозащищенностью (благодаря металлической оплетке), а также более высокими, чем в случае витой пары, допустимыми скоростями передачи данных (до 500 Мбит/с) и большими допустимыми расстояниями передачи (до километра и выше). К нему труднее механически подключиться для несанкционированного прослушивания сети, он также дает заметно меньше электромагнитных излучений вовне. Однако монтаж и ремонт коаксиального кабеля существенно сложнее, чем витой пары, а стоимость его выше (он дороже примерно в 1,5 – 3 раза по сравнению с кабелем на основе витых пар). Сложнее и установка разъемов на концах кабеля. Поэтому его сейчас применяют реже, чем витую пару.

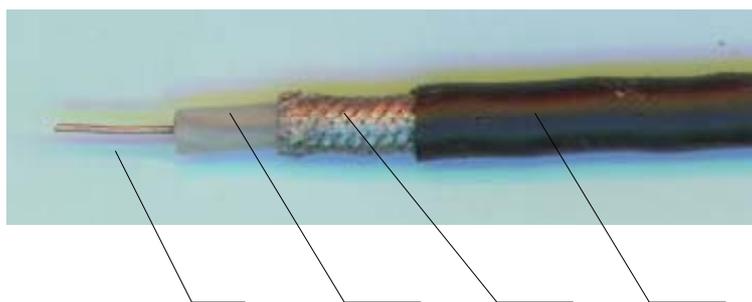


Рис. 3.26. Коаксиальный кабель

Основное применение коаксиальный кабель находит в сетях с топологией типа «шина». При этом на концах кабеля обязательно должны устанавливаться терминаторы для предотвращения внутренних отражений сигнала, причем один (и только один!) из терминаторов должен быть заземлен. Без заземления металлическая оплетка не защищает сеть от внешних электромагнитных помех и не снижает излучение передаваемой по сети информации во внешнюю среду. Но при заземлении оплетки в двух или более точках из строя может выйти не только сетевое оборудование, но и компьютеры, подключенные к сети.

Терминаторы должны быть обязательно согласованы с кабелем, то есть их сопротивление должно быть равно волновому сопротивлению кабеля. Например, если используется 50-омный кабель, для него подходят только 50-омные терминаторы. Реже коаксиальные кабели применяются в сетях с топологиями «звезда» и «пассивная звезда» (например, в сети

Arcnet). В этом случае проблема согласования существенно упрощается, так как внешних терминаторов на свободных концах не требуется. Волновое сопротивление кабеля указывается в сопроводительной документации. Чаще всего в локальных сетях применяются 50-омные (например, RG-58, RG-11) и 93-омные кабели (например, RG-62). 75-омные кабели, распространенные в телевизионной технике, в локальных сетях не используются. Вообще, марок коаксиального кабеля значительно меньше, чем кабелей на основе витых пар. Он не считается особо перспективным. Не случайно в сети Fast Ethernet не предусмотрено применение коаксиальных кабелей. Однако во многих случаях классическая шинная топология (а не топология «пассивная звезда») очень удобна. Как уже отмечалось, она не требует применения дополнительных устройств – концентраторов.

Существует два основных типа коаксиального кабеля: тонкий (thin) кабель, имеющий диаметр около 0,5 см, более гибкий; толстый (thick) кабель, имеющий диаметр около 1 см, значительно более жесткий, он представляет собой классический вариант коаксиального кабеля, который уже почти полностью вытеснен более современным тонким кабелем. Тонкий кабель используется для передачи на меньшие расстояния, чем толстый, так как в нем сигнал затухает сильнее, но зато с ним гораздо удобнее работать: его можно оперативно проложить к каждому компьютеру, а толстый требует жесткой фиксации на стене помещения. Подключение к тонкому кабелю (с помощью разъемов BNC байонетного типа) проще и не требует дополнительного оборудования, а для подключения к толстому кабелю надо использовать специальные довольно дорогие устройства, прокалывающие его оболочки и устанавливающие контакт как с центральной жилой, так и с экраном. Толстый кабель примерно вдвое дороже тонкого. Поэтому тонкий кабель применяется гораздо чаще.

Как и в случае витых пар, важным параметром коаксиального кабеля является тип его внешней оболочки. Точно так же в данном случае применяются как non-plenum (PVC), так и plenum кабели. Естественно, тефлоновый кабель дороже поливинилхлоридного. Обычно тип оболочки можно отличить по ее окраске (например, для кабеля PVC фирма Belden использует желтый цвет, а для тефлонового – оранжевый). Типичные величины задержки распространения сигнала в коаксиальном кабеле составляют для тонкого кабеля около 5 нс/м, а для толстого – около 4,5 нс/м. Существуют варианты коаксиального кабеля с двойным экраном (один экран расположен внутри другого и отделен от него дополнительным слоем изоляции). Такие кабели имеют лучшую помехозащищенность и защиту от прослушивания, но они немного дороже обычных.

В настоящее время считается, что коаксиальный кабель устарел, в большинстве случаев его вполне может заменить витая пара или оптоволоконный кабель. Новые стандарты на кабельные системы уже не включают его в перечень типов кабелей.

Контрольные вопросы

1. Дайте определение физической среды передачи данных и перечислите виды кабелей.
2. Приведите сравнительные характеристики физических сред и охарактеризуйте области их возможных применений.
3. Что представляет собой кабель «витая пара»? Какими параметрами он характеризуется?
4. Дайте представление о горизонтальных кабелях типа «витая пара» и их конструктивных особенностях.
5. Экранированный и неэкранированный кабель «витая пара». Назначение экрана.
6. Что означает понятие «категория кабеля» и чем отличаются друг от друга кабели различных категорий?
7. Назначение магистральных кабелей, их основные параметры, маркировка, конструктивное исполнение.
8. В каких случаях применяют конструкции кабелей S/STP, STP, S/UTP?
9. На какие параметры влияет шаг скрутки витой пары?
10. За счет чего обеспечивается снижение электромагнитных помех в витой паре?
11. От чего зависит скорость распространения сигнала и какой величиной нормируется?
12. Как проявляется переходное затухание и затухание погонное?
13. Принцип действия оптического кабеля.
14. Назовите виды дисперсий и прокомментируйте их влияние на пропускную способность канала.
15. Дайте понятие моды.
16. Одно и многомодовое волокно, характеристики, особенности работы.
17. От чего зависит затухание сигнала в оптическом канале и что такое «окно прозрачности»?
18. Перечислите геометрические параметры оптического кабеля и укажите, как они влияют на прохождение сигнала.
19. Что такое коннектор, для чего используется и какие виды коннекторов бывают?
20. Что представляют собой MT- и MTP-коннекторы, их конструктивное исполнение.

Глава 4. ПОВТОРИТЕЛИ, КОММУТАТОРЫ

4.1. Техническая реализация и дополнительные функции коммутаторов

Несмотря на то, что в коммутаторах работают известные и хорошо отработанные алгоритмы прозрачных мостов и мостов с маршрутизацией от источника, существует большое разнообразие моделей коммутаторов. Они отличаются как внутренней организацией, так и набором выполняемых дополнительных функций, таких как трансляция протоколов, поддержка алгоритма покрывающего дерева, образование виртуальных логических сетей и ряда других.

После того как технология коммутации привлекла общее внимание и получила высокие оценки специалистов, многие компании занялись реализацией этой технологии в своих устройствах, применяя для этого различные технические решения. Многие коммутаторы первого поколения были похожи на маршрутизаторы, то есть основывались на центральном процессоре общего назначения, связанном с интерфейсными портами по скоростной внутренней шине (рис. 4.1.). Однако это были скорее пробные устройства, предназначенные для освоения самой компанией технологии коммутации, а не для завоевания рынка.

Основным недостатком таких коммутаторов была их низкая скорость. Универсальный процессор никак не мог справиться с большим объемом специализированных операций по пересылке кадров между интерфейсными модулями.

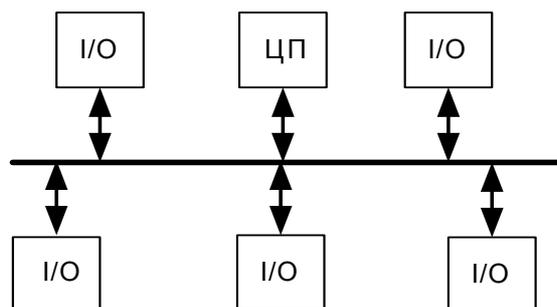


Рис. 4.1. Коммутатор на процессоре общего назначения

Для ускорения операций коммутации нужны были специализированные процессоры со специализированными средствами обмена данными, как в первом коммутаторе Kalpana, и они вскоре появились. Сегодня все коммутаторы используют заказные специализированные БИС – ASIC, которые оптимизированы для выполнения основных операций коммутации. Часто в одном коммутаторе используются несколько специализированных ИС, каждая из которых выполняет функционально законченную часть операций.

Сравнительно низкая стоимость современных коммутаторов по сравнению с их

предшественниками 3 – 5-летней давности объясняется массовым характером производства основных БИС, на которых каждая компания строит свои коммутаторы.

Кроме процессорных микросхем для успешной неблокирующей работы коммутатору нужно также иметь быстродействующий узел для передачи кадров между процессорными микросхемами портов.

В настоящее время коммутаторы используют в качестве базовой одну из трех схем, на которой строится такой узел обмена:

- коммутационная матрица;
- разделяемая многовходовая память;
- общая шина.

Часто эти три способа взаимодействия комбинируются в одном коммутаторе.

4.2. Коммутаторы на основе коммутационной матрицы

Коммутационная матрица обеспечивает основной и самый быстрый способ взаимодействия процессоров портов, именно он был реализован в первом промышленном коммутаторе локальных сетей. Однако реализация матрицы возможна только для определенного числа портов, причем сложность схемы возрастает пропорционально квадрату количества портов коммутатора (рис. 4.2).

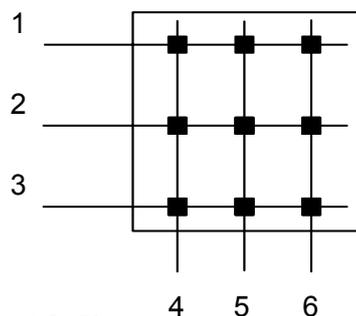


Рис. 4.2. Коммутационная матрица

Более детальное представление одного из возможных вариантов реализации коммутационной матрицы для 8 портов дано на рис. 4.3. Входные блоки процессоров портов на основании просмотра адресной таблицы коммутатора определяют по адресу назначения номер выходного порта. Эту информацию они добавляют к байтам исходного кадра в виде специального ярлыка – тэга (tag). Для данного примера тэг представляет собой просто 3-разрядное двоичное число, соответствующее номеру выходного порта.

Матрица состоит из трех уровней двоичных переключателей, которые соединяют свой вход с одним из двух выходов в зависимости от значения бита тэга. Переключатели первого уровня управляются первым битом тэга, второго – вторым, а третьего – третьим.

Матрица может быть реализована и по-другому, на основании комбинационных схем другого типа, но ее особенностью все равно остается технология коммутации физических каналов. Известным недостатком этой технологии является отсутствие буферизации данных внутри коммутационной матрицы: если составной канал невозможно построить из-за занятости выходного порта или промежуточного коммутационного элемента, то данные должны накапливаться в их источнике, в данном случае во входном блоке порта, принявшего кадр. Основные достоинства таких матриц – высокая скорость коммутации и регулярная структура, которую удобно реализовывать в интегральных микросхемах. Зато после реализации матрицы $N \times N$ в составе БИС проявляется еще один ее недостаток – сложность наращивания числа коммутируемых портов.

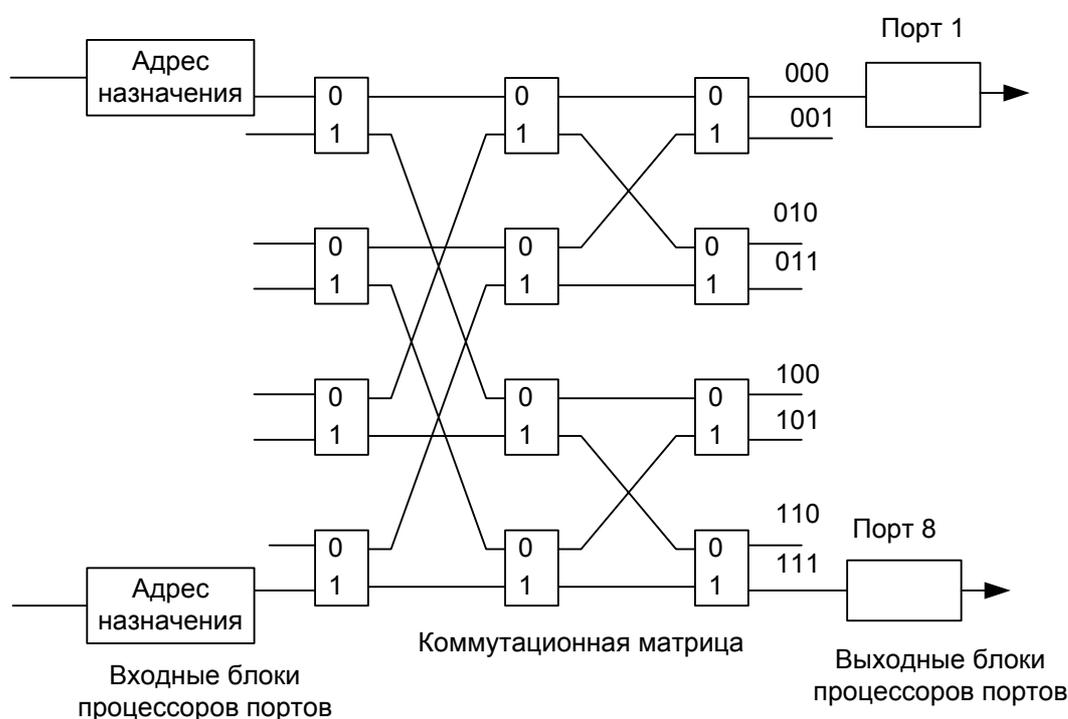


Рис. 4.3. Реализация коммутационной матрицы 8×8 с помощью двоичных переключателей

4.3. Коммутаторы с общей шиной

В коммутаторах с общей шиной процессоры портов связаны высокоскоростной шиной, используемой в режиме разделения времени.

Пример такой архитектуры приведен на рис. 4.4. Чтобы шина не блокировала работу коммутатора, ее производительность должна равняться, по

крайней мере, сумме производительности всех портов коммутатора. Для модульных коммутаторов некоторые сочетания модулей с низкоскоростными портами могут приводить к неблокирующей работе, а установка модулей с высокоскоростными портами может приводить к тому, что блокирующим элементом станет, например, общая шина.

Кадр должен передаваться по шине небольшими частями, по несколько байт, чтобы передача кадров между несколькими портами происходила в псевдопараллельном режиме, не внося задержек в передачу кадра в целом. Размер такой ячейки данных определяется производителем коммутатора. Некоторые производители, например LANNET или Centillion, выбрали в качестве порции данных, переносимых за одну операцию по шине, ячейку ATM с ее полем данных в 48 байт. Такой подход облегчает трансляцию протоколов локальных сетей в протокол ATM, если коммутатор поддерживает эти технологии.

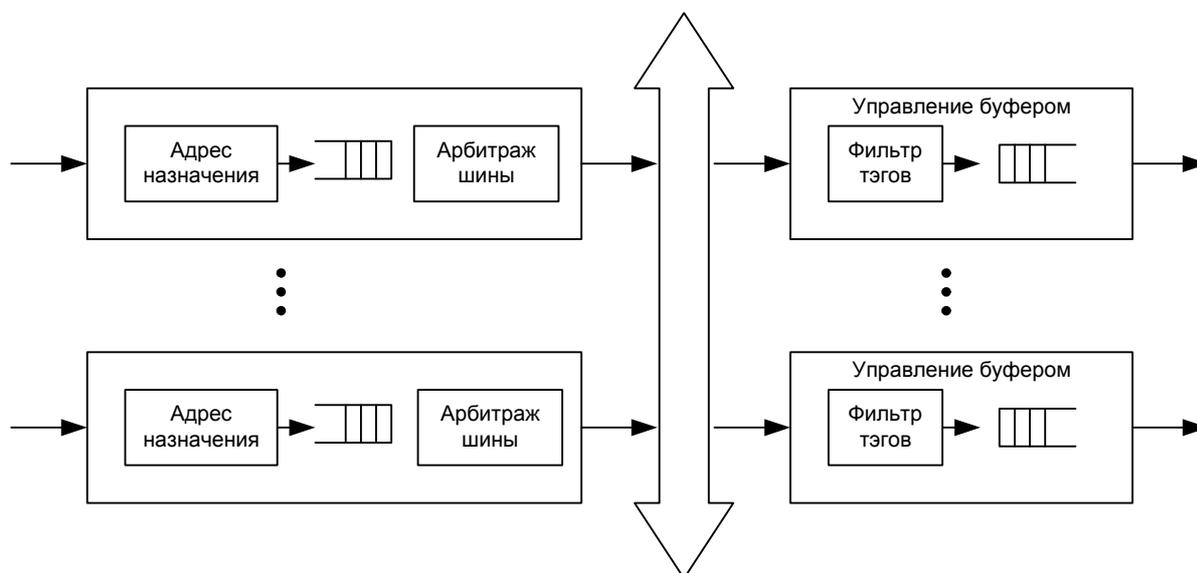


Рис. 4.4. Архитектура коммутатора с общей шиной

Входной блок процессора помещает в ячейку, переносимую по шине, тэг, в котором указывает номер порта назначения. Каждый выходной блок процессора i -го порта содержит фильтр тэгов, который выбирает тэги, предназначенные данному порту.

Шина, так же как и коммутационная матрица, не может осуществлять промежуточную буферизацию, но так как данные кадра разбиваются на небольшие ячейки, то задержек с начальным ожиданием доступности выходного порта в такой схеме нет (здесь работает принцип коммутации пакетов, а не каналов).

4.4. Коммутаторы с разделяемой памятью

Третья базовая архитектура взаимодействия портов – двухвходовая разделяемая память. Пример такой архитектуры приведен на рис. 4.5.

Входные блоки процессоров портов соединяются с переключаемым входом разделяемой памяти, а выходные блоки этих же процессоров соединяются с переключаемым выходом этой памяти. Переключением входа и выхода разделяемой памяти управляет менеджер очереди выходных портов. В разделяемой памяти менеджер организует несколько очередей данных, по одной для каждого выходного порта. Входные блоки процессоров передают менеджеру портов запросы на запись данных в очередь того порта, который соответствует адресу назначения пакета. Менеджер по очереди подключает вход памяти к одному из входных блоков процессоров, и тот переписывает часть данных кадра в очередь определенного выходного порта. По мере заполнения очередей менеджер производит также поочередное подключение выхода разделяемой памяти к выходным блокам процессоров портов, и данные из очереди переписываются в выходной буфер процессора.

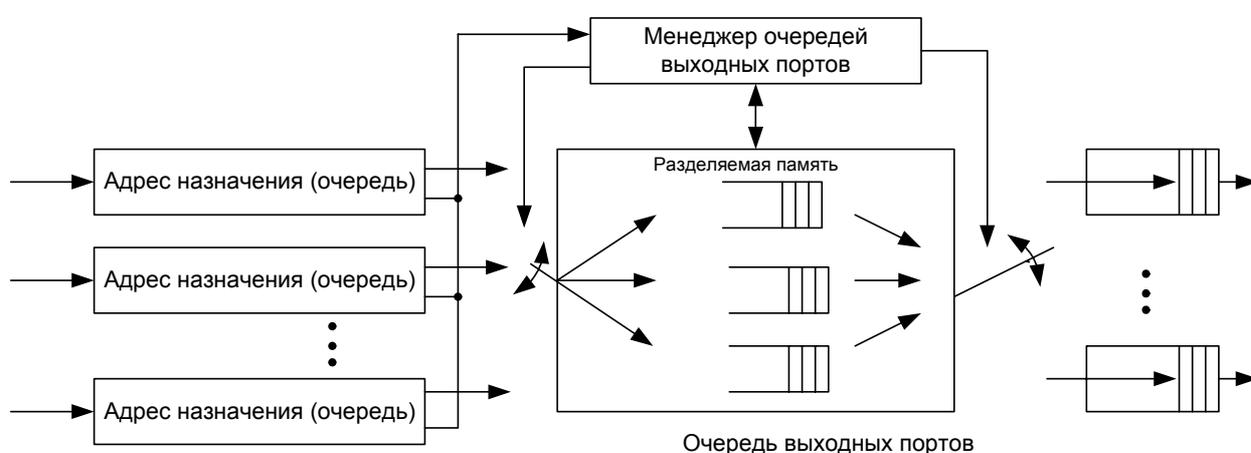


Рис. 4.5. Архитектура разделяемой памяти

Память должна быть достаточно быстрой для поддержания скорости переписи данных между N портами коммутатора. Применение общей буферной памяти, гибко распределяемой менеджером между отдельными портами, снижает требования к размеру буферной памяти процессора порта.

4.5. Комбинированные коммутаторы

У каждой из описанных архитектур есть свои преимущества и недостатки, поэтому часто в сложных коммутаторах эти архитектуры применяются в комбинации друг с другом. Пример такого комбинирования приведен на рис. 4.6.

Коммутатор состоит из модулей с фиксированным количеством портов (2 – 12), выполненных на основе специализированной БИС, реализующей архитектуру коммутационной матрицы. Если порты, между которыми нужно передать кадр данных, принадлежат одному модулю, то передача кадра осуществляется процессорами модуля на основе имеющейся в модуле коммутационной матрицы. Если же порты принадлежат разным моду-

лям, то процессоры общаются по общей шине. При такой архитектуре передача кадров внутри модуля будет происходить быстрее, чем при межмодульной передаче, так как коммутационная матрица – наиболее быстрый, хотя и наименее масштаби-

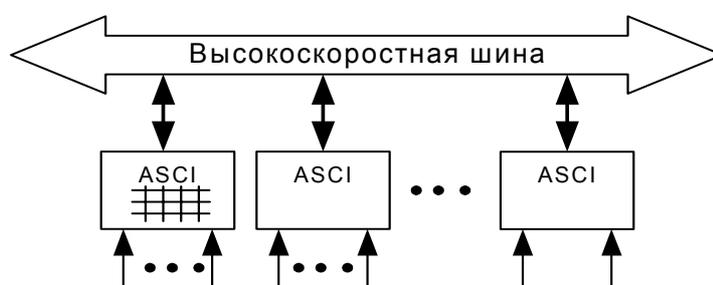


Рис. 4.6. Комбинированный коммутатор

руемый способ взаимодействия портов. Скорость внутренней шины коммутаторов может достигать нескольких гигабит в секунду, а у наиболее мощных моделей – до 20 – 30 Гбит/с.

Можно представить и другие способы комбинирования архитектур, например использование разделяемой памяти для взаимодействия модулей.

4.6. Конструктивное исполнение коммутаторов

В конструктивном отношении коммутаторы делятся на следующие типы:

- автономные коммутаторы с фиксированным количеством портов;
- модульные коммутаторы на основе шасси;
- коммутаторы с фиксированным количеством портов, собираемые в стек.

Первый тип коммутаторов обычно предназначен для организации небольших рабочих групп.

Модульные коммутаторы на основе шасси чаще всего предназначены для применения на магистрали сети. Поэтому они выполняются на основе какой-либо комбинированной схемы, в которой взаимодействие модулей организуется по быстродействующей шине или же на основе быстрой разделяемой памяти большого объема. Модули такого коммутатора выполняются на основе технологии «hot swap», то есть допускают замену на ходу, без выключения коммутатора, так как центральное коммуникационное устройство сети не должно иметь перерывов в работе. Шасси обычно снабжается резервированными источниками питания и резервированными вентиляторами в тех же целях.

С технической точки зрения определенный интерес имеют стековые коммутаторы. Эти устройства представляют собой коммутаторы, которые могут работать автономно, так как выполнены в отдельном корпусе, но имеют специальные интерфейсы, которые позволяют их объединять в общую систему, работающую как единый коммутатор. Говорят, что в этом случае отдельные коммутаторы образуют стек.

Обычно такой специальный интерфейс представляет собой высокоскоростную шину, которая позволяет объединить отдельные корпуса подобно модулям в коммутаторе на основе шасси. Так как расстояния между корпусами больше, чем между модулями на шасси, скорость обмена по шине обычно ниже, чем у модульных коммутаторов: 200 – 400 Мбит/с. Не очень высокие скорости обмена между коммутаторами стека обусловлены также тем, что стековые коммутаторы обычно занимают промежуточное положение между коммутаторами с фиксированным количеством портов и коммутаторами на основе шасси. Стековые коммутаторы применяются для создания сетей рабочих групп и отделов, поэтому сверхвысокие скорости шин обмена им не очень нужны и не соответствуют их ценовому диапазону.

4.7. Репитеры

При распространении сигнала вдоль линии связи его форма искажается: «заваливаются» фронты, уменьшается амплитуда, появляются «наложения» из-за влияния внешних помех, отражений и т.д. Для восстановления формы сигнала используются ретрансляторы, в простейшем случае – обычные логические элементы ТТЛ с повышенной нагрузочной способностью и

желательно с гистерезисной передаточной характеристикой вход – выход. Сигнал с входа A (рис. 4.7, a) проходит через элемент $D1$, линию связи $L1$, элемент-ретранслятор $D2$, линию связи $L2$, выходной элемент $D3$ и поступает на выход B .

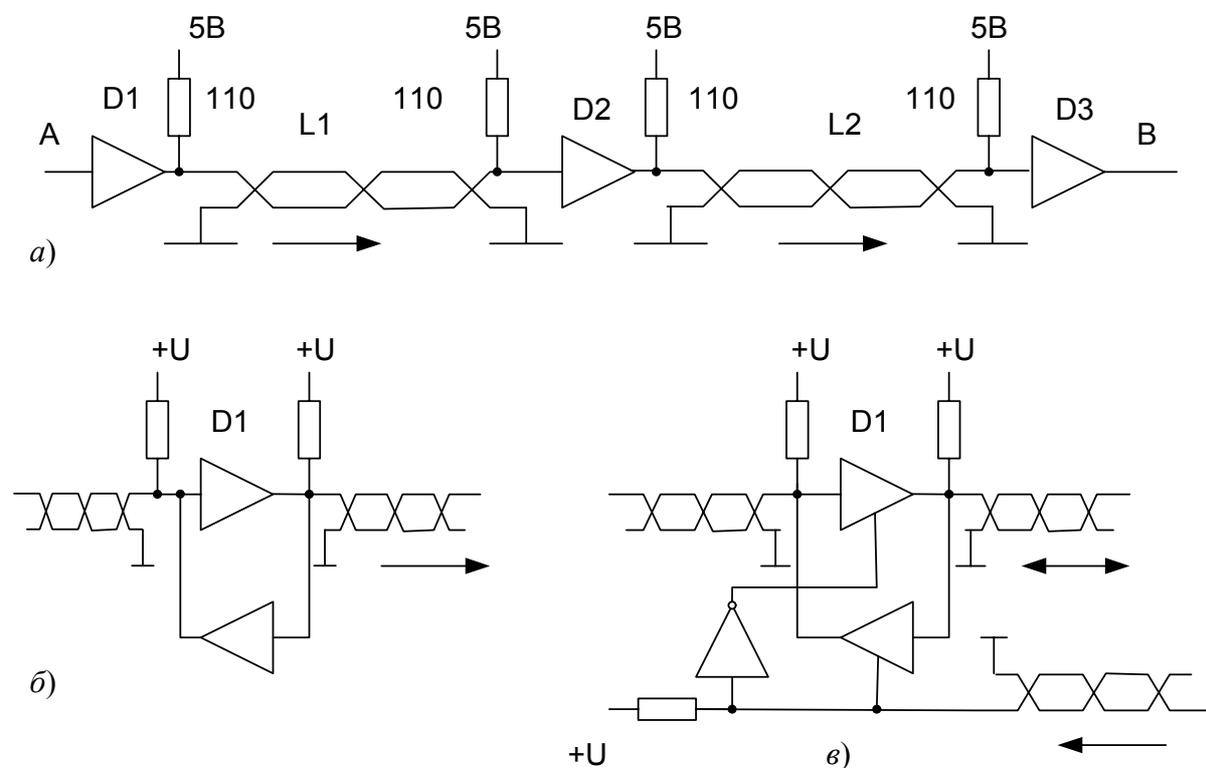


Рис. 4.7. Схемы повторителей (ретрансляторов) сигналов:

a – использование магистральных логических элементов ТТЛ для однонаправленной передачи сигналов; $б$ – неудачная попытка использования двух встречно включенных логических элементов для двунаправленной ретрансляции сигналов; $в$ – неудачная попытка использования управляющего сигнала Y

Резисторы, включенные на концах отрезков линий связи, служат нагрузками для выходных элементов с открытым коллектором и предотвращают отражения сигналов от концов линии. Обычно вместо отдельных резисторов используют резисторные делители напряжения, например $180/390$ Ом (этот прием позволяет соблюдать условия согласования линии при меньших энергетических затратах).

При однонаправленной передаче сигналов допустимо включение согласующего резистора или резисторного делителя напряжения только со стороны входа элемента – приемника. Однако в этом случае на коллекторе передающего транзистора могут наблюдаться положительные выбросы напряжения

тем большие, чем хуже выполнены условия согласования линии на приемном конце, а именно чем ближе эти условия к короткому замыканию. Так что, может быть, лучше перестраховаться и согласовать однонаправленную линию с двух концов, а не с одного.

Ретрансляция двунаправленных сигналов, к сожалению, не может производиться столь же простыми средствами. Действительно, если включить два логических элемента параллельно, но навстречу друг другу, то они образуют триггер (рис. 4.7, б), который, однажды оказавшись в состоянии, при котором $A = B = 0$, будет оставаться в нем неограниченно долгое время, вплоть до момента снятия напряжения питания.

Использование управляющего сигнала Y (рис. 4.7, в), который запрещает передачу информации D в обе стороны одновременно, не решает проблемы, поскольку для передачи сигнала Y нужна дополнительная линия связи, а мы как раз и стремимся сократить их число. Рассмотрим два наиболее экономичных варианта схем двунаправленных приемопередатчиков, не требующих внешнего управляющего сигнала для указания направления передачи. На рис. 4.8 показана одна из таких схем [3]. В исходном состоянии в точках A и B поддерживаются высокие (+ 3,5 В) уровни напряжения, на выходах элементов «ИЛИ – НЕ» сформированы лог. «0». Инверторы с открытым коллектором, выполняющие роль передатчиков ретранслятора, выключены.

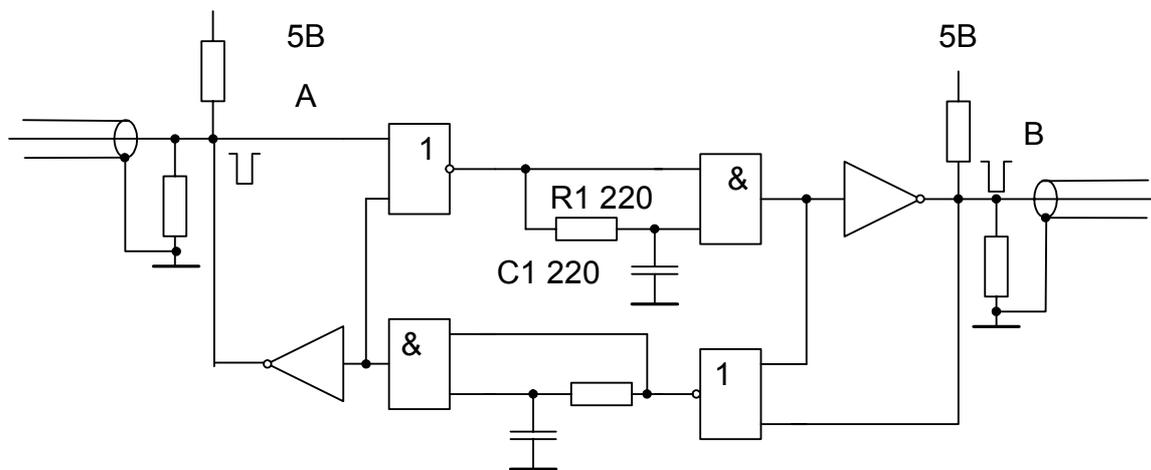


Рис. 4.8. Двунаправленный ретранслятор (первый вариант)

Отрицательный фронт импульса с входа A после инвертирования поступает на входы элемента «И» и затем через выходной инвертор проходит на выход B . Цепь $R1C1$ в данном случае не выполняет полезных функций и вносит нежелательную задержку при прохождении сигнала с входа A на выход B .

При окончании отрицательного импульса в точке A (в точке A – напряжение высокого уровня) сигнал, минуя цепь $R1C1$, проходит через элемент

«И» и инвертор, вызывая повышение напряжения в точке B до $+3,5$ В. Поскольку задержка передачи сигнала через выходной инвертор, подключенный к точке B , не равна нулю и сильно зависит от паразитной емкости выхода, при передаче среза импульса возникает ситуация, при которой на входе и выходе этого инвертора одновременно присутствуют лог. «0». Это приводит к появлению положительного кратковременного импульса на выходе элемента «ИЛИ – НЕ» цепи обратной передачи сигнала.

Если не принять мер по его подавлению, то сработает инвертор, подключенный к точке A , и затем импульс помехи будет распространяться по кольцу, то есть возникнет генерация. Подавление помехи производится цепью $R2C2$. Ее параметры выбирают так, чтобы к моменту окончания импульса на верхнем входе элемента «И» конденсатор $C2$ не успел зарядиться до порогового напряжения этого элемента.

Передача полезного сигнала (отрицательного импульса) в противоположном направлении от B к A производится аналогично. В этом случае помеха на срезе подавляется цепью $R1C1$. Поэтому недостаток схемы заключается в необходимости подбора RC -цепей для каждого конкретного случая ее применения.

Более совершенной (но и более сложной) является схема, приведенная на рис. 4.9.

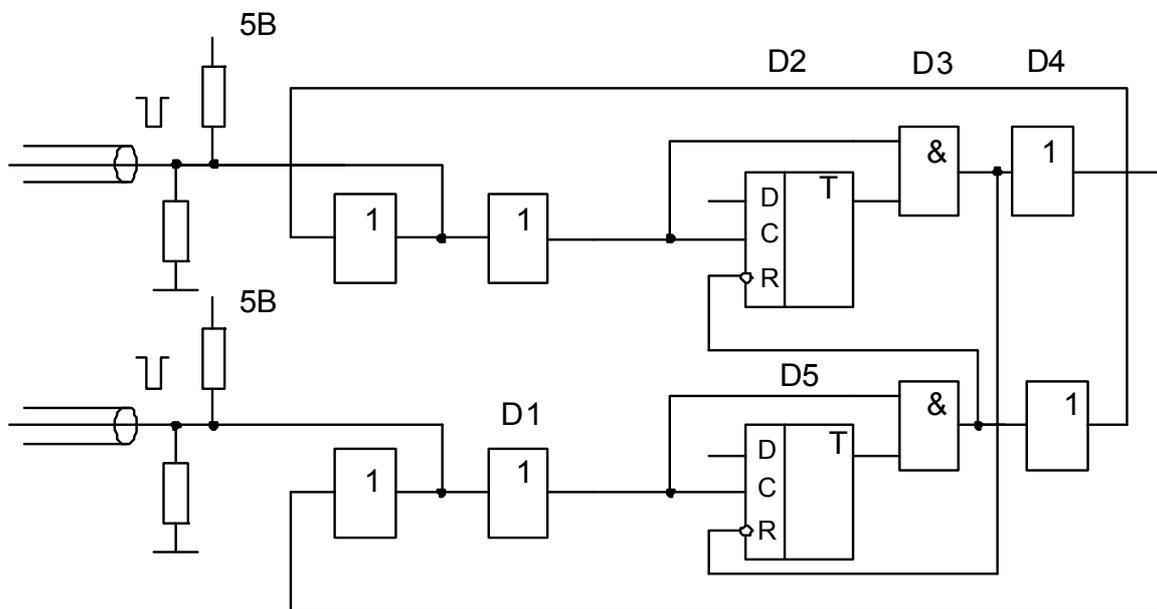


Рис. 4.9. Двухнаправленный ретранслятор (второй вариант)

По отрицательному фронту сигнала на входе A триггер $D1$ устанавливается в 1, открывается элемент $D2$ и сигнал передается через элементы $D3$ и $D4$ в канал B . Сигнал лог. «0» с выхода элемента $D2$ поддерживает триггер

гер D5 в нулевом состоянии, размыкая обратную связь, поэтому по окончании импульса никаких конфликтов не возникнет. Если триггер D1 первоначально уже был установлен в 1, то задержка передачи сигнала уменьшается на время срабатывания триггера. Это при многократной ретрансляции может привести к заметным вариациям длительности передаваемых импульсов.

Выходные элементы ретранслятора выполнены по схеме с открытым коллектором; резисторные делители служат в качестве нагрузок для выходных инверторов и обеспечивают согласование с кабелями по волновым сопротивлениям. Передача отрицательного импульса из *B* в *A* производится аналогично.

Схему рис. 4.9 можно усовершенствовать с тем, чтобы ее задержка не зависела от состояния триггеров D1 и D5 к моменту поступления входного импульса. Для этого можно ввести два элемента задержки в цепи верхних входов элементов «И – НЕ». Каждый из этих элементов задержки будет препятствовать преждевременному прохождению сигнала «в обход» соответствующего триггера, то есть задержка каждого из введенных элементов выбирается равной или большей, чем задержка триггера. Возможны и иные решения. Например, можно устанавливать оба триггера в 1 в момент окончания передачи отрицательного импульса, а также после включения напряжения питания (чтобы условия распространения первого и последующих импульсов были одинаковыми).

Схемы ретрансляторов, приведенные на рис. 4.8 и 4.9, работают «правильно» только при отсутствии конфликтов между импульсами. Иными словами, ретранслятор передает импульсы в одном или разных направлениях без искажений, если эти импульсы разделены во времени паузами. Если встречные импульсы частично накладываются друг на друга (это обычная ситуация для линии, реализующей функцию монтажного «И» («ИЛИ»)), то работа схем оказывается довольно своеобразной, если не сказать «неправильной», так как практически любое «своеобразие» можно с пользой применить.

На рис. 4.10 приведены временные диаграммы, соответствующие работе схем при наложении импульсов. Передатчики, подключенные к линии, выполнены по схеме с открытым коллектором.

Функция монтажного «И» (C&D) «в чистом виде» реализуется только на отрезке линии, связанном с точкой *B*, так как в данном примере сигнал *D* заканчивается после окончания сигнала *C*.

«Двунаправленность» теряется еще в большей степени, чем при использовании ретранслятора, приведенного на рис. 4.8.

Отрезки линий согласованы на концах (согласующие резисторы не показаны). Обе схемы (см. рис. 4.8 и 4.9.) не гарантируют правильную реализацию функции монтажного «И», если заранее не известна последовательность формирования встречных импульсов (а так чаще всего и бывает). Еще более причудливые ситуации возникают при рассмотрении цепей из нескольких ретрансляторов.

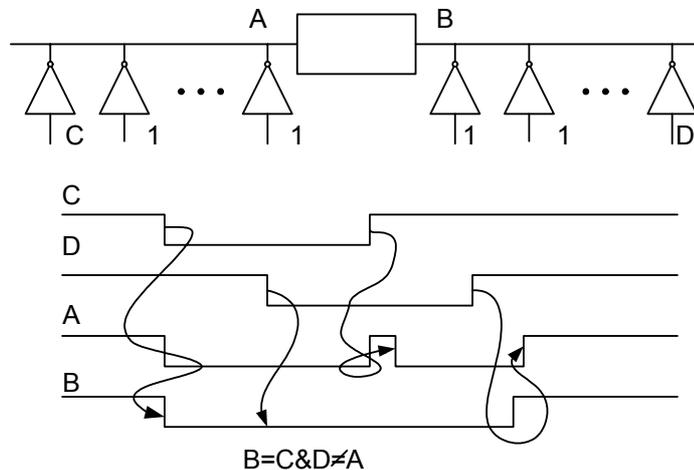


Рис. 4.10. Работа схемы (см. рис. 4.8) при наложении сигналов *C* и *D*

Таким образом, проблема построения системы ретрансляторов, функционально эквивалентной обычному проводу, не так проста, как это может показаться на первый взгляд.

Контрольные вопросы

1. Назовите базовые схемы для построения коммутаторов и дайте их краткую характеристику.
2. По какому принципу строится коммутационная матрица?
3. Опишите принцип работы коммутатора с общей шиной и укажите на особенности работы общей шины в процессе передачи информации.
4. Роль разделяемой памяти в коммутаторах и роль менеджера при коммутации портов.
5. Укажите особенности работы комбинированных коммутаторов.
6. Для каких целей в коммутаторах используется стек?
7. Каково назначение репитеров?
8. Роль репитеров в передаче информации.
9. Приведите пример схмотехнического решения двунаправленного репитера функционально эквивалентному проводу.

Глава 5. ПОСЛЕДОВАТЕЛЬНЫЕ ИНТЕРФЕЙСЫ RS-232 и RS-485

При построении систем управления с использованием сетевых технологий в качестве управляющей ЭВМ используется платформа IBM PC. Структурная схема такой системы представлена на рис. 5.1 и состоит из управляющей ЭВМ и комплекса малых ЭВМ, связанных между собой последовательным каналом.

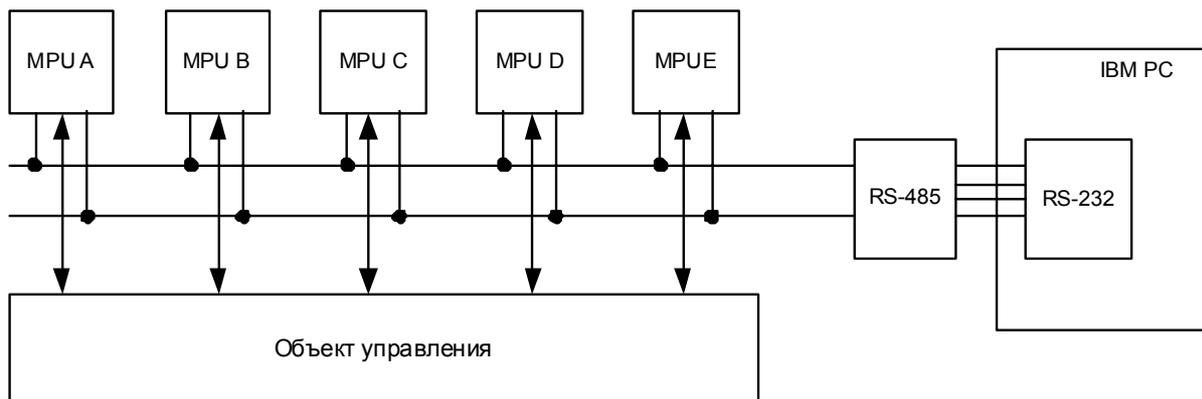


Рис. 5.1. Структурная схема локальной сети контроллеров

Одним из коммуникационных портов, входящих в структуру IBM PC, является локальный последовательный порт RS-232. Связь через последовательный асинхронный порт во многих применениях еще носит актуальный характер, несмотря на появившиеся системы связи, являющиеся более эффективными. Сам по себе этот порт позволяет соединять только два устройства, разнесенные на небольшое расстояние. Подключение большего числа устройств требует иного соединения, в качестве которого используются системы двухпроводной связи, например интерфейс RS-485.

В состав малых ЭВМ, построенных на основе микроконтроллеров не входят интерфейсы USB, IEEE1394, CAN. Однако сложно найти контроллер, не имеющий в составе своей периферии UART.

Наиболее широко используются сети, построенные на основе интерфейсов RS-232 и RS-485. Рассмотрим их.

Описание конкретной реализации последовательного порта есть в описании на каждый контроллер, но у каждого из них есть общее для всех.

Универсальный асинхронный приемопередатчик (UART) можно представить приемником (receiver) и передатчиком (transmitter). В состав UART

входят: тактовый генератор связи (бодрейт-генератор), управляющие регистры, статусные регистры, буферы и сдвиговые регистры приемника и передатчика. Бодрейт-генератор задает тактовую частоту приемопередатчика для данной скорости связи. Управляющие регистры задают режим работы последовательного порта и его прерываний. В статусном регистре устанавливаются флаги по различным событиям. В буфер приемника попадает принятый символ, в буфер передатчика помещают передаваемый. Сдвиговый регистр передатчика – это обойма, из которой в последовательный порт передаются биты передаваемого символа (кадра). Сдвиговый регистр приемника по биту накапливает принимаемые из порта биты. По различным событиям устанавливаются флаги и генерируются прерывания (завершение приема/отправки кадра, освобождение буфера, различные ошибки).

UART – полнодуплексный интерфейс, то есть приемник и передатчик могут работать одновременно, независимо друг от друга. За каждым из них закреплен порт – одна ножка контроллера. Порт приемника обозначают RX, передатчика – TX. Последовательной установкой уровней на этих портах относительно общего провода ("земли") и передается информация. По умолчанию передатчик устанавливает на линии единичный уровень. Передача начинается посылкой бита с нулевым уровнем (старт-бита), затем идут биты данных младшим битом вперед (низкий уровень – "0", высокий уровень – "1"), завершается посылка передачей одного или двух битов с единичным уровнем (стоп-битов).

Электрический сигнал кадра посылки выглядит так, как представлено на рис. 5.2.

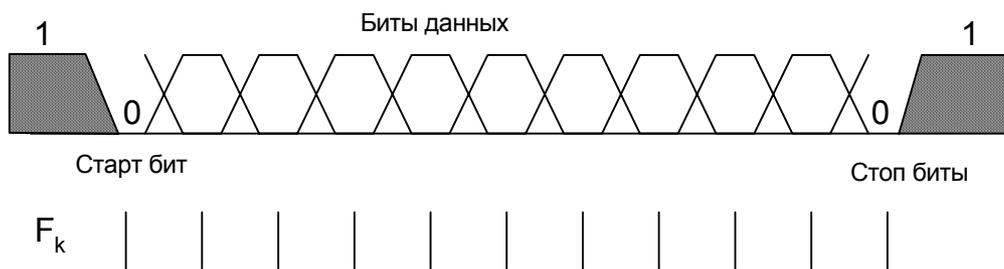


Рис. 5.2. Содержание кадра

Перед началом связи между двумя устройствами необходимо настроить их приемопередатчики на одинаковую скорость связи и формат кадра.

Скорость связи, или baudрейт (baudrate), измеряется в бодах – число передаваемых бит в секунду (включая старт- и стоп-биты). Задается эта скорость в baudрейт-генераторе делением системной частоты на задаваемый коэффициент. Типичный диапазон скоростей 2400 ... 115200 бод.

Формат кадра определяет число стоп-битов (1; 1,5 или 2), число бит данных (8 или 9), а также назначение девятого бита данных. Все это зависит от типа контроллера.

Приемник и передатчик тактируются, как правило, с 16-кратной частотой относительно baudрейта. Это нужно для сэмплирования сигнала. Приемник, поймав спадающий фронт старт-бита, отсчитывает несколько тактов и следующие три такта считывает (семплирует) порт RX. Это как раз середина старт-бита. Если большинство значений семплов "0", старт-бит считается состоявшимся, иначе приемник принимает его за шум и ждет следующего падающего фронта. После удачного определения старт-бита, приемник точно так же семплирует серединки битов данных и по большинству семплов считает бит "0" или "1", записывая их в сдвиговый регистр. Стоп-биты тоже семплируются, и если уровень стоп-бита не "1" – UART определяет ошибку кадра и устанавливает соответствующий флаг в управляющем регистре.

Поскольку baudрейт устанавливается делением системной частоты, при переносе программы на устройство с другим кварцевым резонатором необходимо изменить соответствующие настройки UART.

Последовательный интерфейс предназначен для передачи данных и использует одну сигнальную линию для передачи в одном направлении.

Последовательная передача данных может осуществляться в синхронном и асинхронном режимах. При асинхронной передаче каждому байту данных предшествует старт-бит, за ним следуют биты данных, после них может передаваться бит паритета (четности) и в завершение посылки передается стоп-бит, гарантирующий определенную выдержку времени между соседними посылками (рис. 5.3). Старт-бит следующей посылки может передаваться в любой момент времени, начиная с момента оконча-

ния стоп-бита предыдущей посылки, таким образом, между передачами могут быть паузы произвольной (и неограниченной) длительности.



Рис. 5.3. Семплирование стартового бита

Старт-бит позволяет организовать простую синхронизацию приемника по сигналу от передатчика. При этом подразумевается, что приемник и передатчик работают на одной и той же скорости обмена (то есть имеют одну и ту же тактовую частоту), измеряемой количеством передаваемых бит в секунду. Внутренний генератор синхронизации приемника содержит счетчик-делитель частоты, обнуляемый в момент начала старт-бита от передатчика. Этот счетчик формирует внутренние стробы F_k , по которым приемник фиксирует принимаемые биты. В идеальном случае эти внутренние стробы должны приходиться на середину битовых интервалов, однако из-за разностей скоростей приемника и передатчика возникает накапливающееся рассогласование. Для безошибочного приема 8 бит данных, 1 бита паритета и стоп-бита предельное рассогласование не может превышать 5 % на бит (в идеале строб приходится на середину бита, при отклонении в 50 % мы примем бит с ошибкой, 50 % / 10 бит = 5 %). Очевидно, что чем больше скорость передачи, тем строже требования к синхронизации приемника и передатчика, кроме того, с ростом скорости обмена усиливается влияние фронтов импульсов.

Для асинхронного режима принят ряд стандартных скоростей обмена: 50, 75, 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600 и 115200 бит/с.

Количество бит данных задается программно и может составлять 5, 6, 7 и 8 бит. Количество стоп-бит может быть 1; 1,5; 2 (по длине). Бит паритета может отсутствовать.

Таким образом, для установления связи при асинхронном режиме необходимо установить одинаковыми следующие параметры приемника и передатчика: скорость обмена, количество бит данных, наличие бита паритета, тип паритета (четность – нечетность), длину стоп-бита.

Асинхронный обмен в ПК реализуется с помощью СОМ-порта с использованием протокола RS-232C.

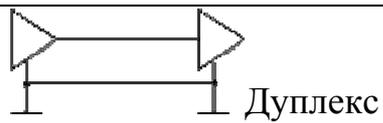
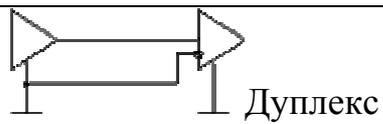
Синхронный режим передачи данных предполагает постоянную активность канала связи. Посылка начинается с передачи синхробайта, за которым вплотную следует последовательность передаваемых бит. Если у передатчика нет данных для передачи, то он заполняет паузу непрерывной передачей старт-бит. Очевидно, что при синхронном обмене можно добиться больших скоростей обмена, так как не требуется предавать старт-бит, стоп-бит и бит паритета (при асинхронном обмене на каждые 8 бит данных приходится 3 – 4 служебных бита). При синхронном обмене необходима внешняя синхронизация, так как передается большой массив данных. При синхронизации только в начале передачи блока данных даже небольшая разность частот приведет к появлению быстро накапливающейся ошибки.

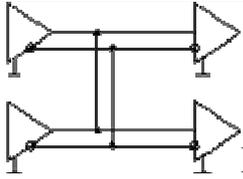
Внешняя синхронизация может обеспечиваться как за счет использования отдельной линии для передачи сигнала синхронизации, так и за счет применения самосинхронизирующихся методов кодирования данных (манчестерский код, NRZ-код), при использовании которых синхросигнал выделяется приемником из потока принимаемых данных.

Существует несколько родственных последовательных интерфейсов: RS-232C, RS-423A, RS-422A, RS-455A. Характеристики этих интерфейсов сведены в табл. 5.1.

Таблица 5.1

Характеристики последовательных интерфейсов

Тип	Схема	V, L
RS-232C	 Дуплекс	$L = 15 \text{ м}, V = 20 \text{ Кбит/с}$
RS-423A	 Дуплекс	$L = 9 \text{ м}, V = 100 \text{ Кбит/с}$ $L = 91 \text{ м}, V = 10 \text{ Кбит/с}$ $L = 1200 \text{ м}, V = 1 \text{ Кбит/с}$

Тип	Схема	V, L
RS-422A	 Дуплекс	$L = 12 \text{ м}, V = 10 \text{ Мбит/с}$ $L = 120 \text{ м}, V = 1 \text{ Мбит/с}$ $L = 1200 \text{ м}, V = 100 \text{ Кбит/с}$
RS-485A	 Полудуплекс, до 32 параллельно соединенных приемопередатчиков	$L = 12 \text{ м}, V = 10 \text{ Мбит/с}$ $L = 120 \text{ м}, V = 1 \text{ Мбит/с}$ $L = 1200 \text{ м}, V = 100 \text{ Кбит/с}$

Линии интерфейсов RS-232C и RS-423A несимметричны, имеют самую низкую защищенность от синфазной помехи. RS-423A имеет приемник с дифференциальным входом, что несколько повышает его помехозащищенность. Лучшими параметрами обладает симметричный дуплексный интерфейс RS-422A и его полудуплексный магистральный аналог RS-485A. Приемник и передатчик RS-422A и RS-485A имеют дифференциальные входы и, следовательно, обладают высокой защищенностью от синфазных помех. Интерфейс RS-485A обладает передатчиком повышенной мощности с защитой от короткого замыкания линии, защитой от перегрева при длительной перегрузке и защитой от коллизий (одновременной работы нескольких параллельно включенных передатчиков).

Одним из вариантов последовательного интерфейса является интерфейс типа "токовая петля". В этом интерфейсе сигналом является не уровень напряжения, а ток в двухпроводной линии. Обычно за единицу принимают ток 20 мА, за ноль – отсутствие тока. В таком варианте интерфейса приемник может распознавать обрыв линии: при обрыве принимаются одни нули и обрыв распознается по отсутствию стоп-битов. Обычно "токовая петля" предполагает наличие гальванической развязки приемника и передатчика, как правило, выполняемой при помощи оптронов.

Питание токовой петли может осуществляться от передатчика (вариант с активным передатчиком) или от приемника (активный приемник). Токовая петля с гальванической развязкой позволяет передавать данные на расстояние до нескольких километров, определяемое уровнем помех и сопротивлением пары проводов. Поскольку интерфейс требует пары проводов для каждого сигнала, то обычно применяют две пары – принимаемые

данные и передаваемые данные, а управление потоком ведется по протоколу XON/XOFF. Одним из классических примеров интерфейса "токовая петля" является интерфейс MIDI, применяемый в звуковых картах.

Все перечисленные выше варианты реализации интерфейса являются проводными, однако существуют и беспроводные варианты, наибольшее распространение среди которых получил инфракрасный (ИК) интерфейс. Большинство ИК-интерфейсов работают на расстоянии от одного до нескольких метров на низкой скорости (до 115,2 Кбит/с), со средней (до 1,152 Мбит/с) или высокой (до 4 Мбит/с) скоростью.

5.1. Интерфейс RS-232C

Интерфейс RS-232C использует несимметричные приемники и передатчики, сигнал передается относительно общего провода (схемной земли). Интерфейс RS-232C не обеспечивает гальванической развязки устройств. Логической единице на входе приемника соответствует уровень напряжения $-3 \dots -12$ В. Для линий управляющих сигналов это состояние называют "ON", а для линий последовательных данных – "MARK". Логическому "0" соответствует напряжение $+3 \dots +12$ В (называемое "OFF" или "SPACE" соответственно). Между уровнями $+3 \dots -3$ В существует зона нечувствительности, обуславливающая гистерезис приемника. Состояние на выходе приемника изменяется только при пересечении напряжением порога $+3$ или -3 В. На рис. 5.4 показан искаженный помехой сигнал, который принимается без ошибки за счет гистерезиса приемника.

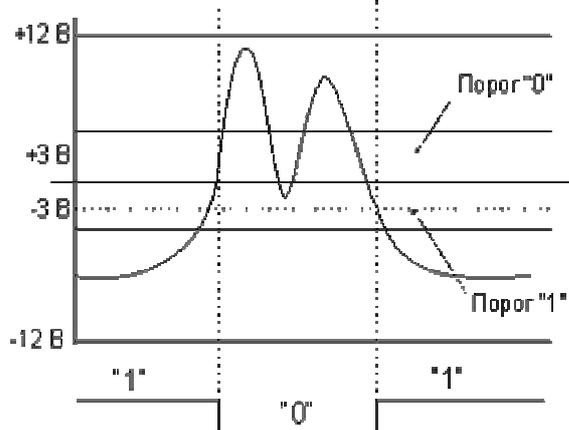


Рис. 5.4. Искаженный сигнал

Уровни сигналов на выходах передатчика должны лежать в диапазоне $+5 \dots +12$ В или $-5 \dots -12$ В.

Интерфейс предполагает наличие защитного заземления обоих устройств.

Присоединение и отключение устройств с автономным питанием

должно производиться при отключенном питании, иначе разность не выровненных потенциалов устройств в момент коммутации может превысить допустимые пределы и вывести из строя микросхемы порта.

Для стандартизации оборудования обычно со стороны DTE (например, в ПК) ставят разъем типа "вилка", со стороны DCE (например, в модеме) – разъем типа "розетка". Тип применяемых разъемов – DB-25S или DB-9S (табл. 5.2).

Таблица 5.2

Линии интерфейса RS-232C

Сигнал	DB9S	Назначение
PG	–	Защитная «земля». Соединяется с корпусом устройства и экраном кабеля
SG	5	Сигнальная «земля» (относительно нее действуют линии сигналов)
TD	3	Выход передатчика
RD	2	Вход приемника
RTS	7	Выход запроса передачи. Состояние "включено" уведомляет модем о том, что у терминала есть данные для передачи. В полудуплексе – переключение модема в режим передачи
CTS	8	Вход разрешения передачи данных терминалу. Состояние "выключено" аппаратно запрещает передачу данных. Применяется для аппаратного управления потоками данных
DTR	4	Выход: "Терминал готов к передаче данных"
DSR	6	Вход сигнала готовности от аппаратуры передачи данных
DCD	1	Вход сигнала обнаружения несущей удаленного модема

Управление потоком данных

Для управления потоком данных могут использоваться два вида протоколов – аппаратный и программный. Не следует путать термин "управление потоком" с термином "квитирование". Квитирование – посылка уведомления о получении элемента, в то время как управление потоком предполагает посылку уведомления о временной невозможности последующего приема данных.

Аппаратный протокол управления RTS/CTS использует сигнал CTS, который позволяет остановить передачу данных, если приемник не готов к работе. Передача данных по этому протоколу показана на рис. 5.5, а. Байт, передаваемый на момент прихода CTS, будет передан, однако с момента окончания его передачи передатчик переходит к ожиданию готовности приемника (то есть снятия CTS). Обеспечивает самую быструю реакцию передатчика на состояние приемника, позволяет организовать обмен, не прибегая к буферизации. Часто используется в принтерах и для соединения компьютеров. В случае с принтером линия CTS ПК должна соединяться с линией RTS принтера, при соединении двух ПК необходимо перекрестное соединение CTS – RTS. Если аппаратный протокол обмена не используется, то на линию CTS ПК необходимо подать сигнал "включено", что обычно достигается соединением CTS ПК с его же RTS перемычкой на разъеме. Аппаратный обмен невозможен через минимальный нуль-модемный кабель.

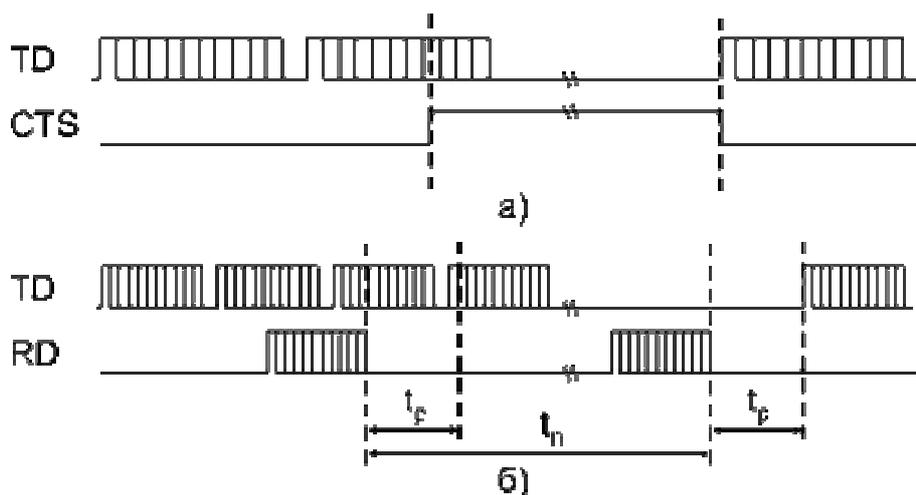


Рис. 5.5. Протокол обмена данными:
 а – аппаратный протокол; б – программный протокол

Аппаратный протокол DTR/DSR. Аналогичен протоколу RTS/CTS, но использует другую пару сигналов (в данном случае "готовность приемника").

Программный протокол XON/XOFF. Предполагает наличие двунаправленного канала обмена (временные диаграммы обмена показаны на рис. 5.5, б) и наличие у приемника буфера, так как время реакции передатчика t_p может оказаться достаточно большим. Когда буфер приемника за-

полняется до определенного уровня (обычно 80 – 90 %), он передает на приемник команду XOFF (байт с кодом 13h). Приняв эту команду, передатчик прекращает передачу и переходит в состояние ожидания до прихода команды XON (байт с кодом 11h), по которому передатчик возобновляет передачу.

Программный протокол АСК. При обмене по этому протоколу для получения очередного байта приемник посылает передатчику команду АСК (байт с кодом 6h). В ответ передатчик посылает приемнику один байт (или пакет байт определенного размера).

Реализация указанных выше протоколов основана на применении микросхем, принцип работы которых заключается в следующем.

Преобразование параллельного кода в последовательный при передаче и последовательного в параллельный при приеме, контроль ошибок, формирование запросов прерывания и ряд других сервисных функций осуществляется специализированной микросхемой, называемой UART (Universal Asynchronous Receiver Transmitter – универсальный асинхронный приемопередатчик). COM-порты ИВР РС базируются на микросхемах 8250/16450/16450A, совместимых на уровне регистров с UART i8250. Каждая из перечисленных микросхем совместима с предыдущей, но не наоборот. Это следует учитывать при составлении программы управления: если программа ориентирована на i8250, то она будет работать со всеми последующими модификациями УАПП. В ряде отечественных ЭВМ применялась микросхема КР580ВВ51 – аналог i8251, так называемый УСАПП (USART) – универсальный синхронно-асинхронный приемопередатчик. Эта микросхема не совместима на уровне регистров с i8250, однако в современных ПК она не применяется (подробнее см. [4]).

Сравнительные характеристики УАПП, совместимых с i8250, приведены в табл. 5.3. Все микросхемы серии 8250 обладают низким быстродействием и не допускают обращения к своим регистрам в смежных циклах процессора – для корректной работы между обращениями организуют программные задержки (команда типа JMP \$+2).

Характеристики УАПП, совместимых с i8250

УАПП	Характеристика
8250	Считается за базовую, самая примитивная по возможностям. Имеет ошибки, приводящие к возможности появления ложных прерываний
8250А	Модернизированная 8050, ошибка устранена, непригоден в АТ для скорости 9600 бит/с, несовместим с XT BIOS
8250В	Исправлены ошибки 8050 и 8050А, совместим с XT BIOS, работает в FN (кроме скорости 9600 бит/с)
16450	Быстродействующая версия 8250 для АТ. Соответствует минимуму для работы с OS/2. Не имеет ошибок, присущих 8250
16550	Развитие 16450, имеет FIFO-буфер (с ошибками, не позволяющими его корректно использовать), может использовать DMA для обмена
16550А	Исправленная версия 16550, имеет нормально работающий 16 байтный FIFO-буфер и DMA, которые должны применяться для работы на скоростях 9600 бит/с и более без потери данных

С 16550А совместимо подавляющее большинство микросхем – контроллеров СОМ-порта современных ПК. Особенностью 16550 и 16550А является то, что они имеют дополнительные, по сравнению с 8250, регистры. Многие биты, считающиеся в 8250 резервом, в 16550 задействованы для управления его новыми функциями. Однако все регистры 8250 совпадают с соответствующими регистрами 16550, что обеспечивает совместимость.

У контроллера последовательного интерфейса имеется 8 регистров, занимающих смежные адреса в пространстве устройств ввода-вывода. Следует обратить особое внимание на то, что регистров реально больше: при чтении регистра «0» байт читается из регистра приемника, при записи байт запишется в регистр передатчика. Кроме того, регистры «0» и «1» мультиплексируются битом «7» регистра управления. Когда этот бит равен 0 (нормальное рабочее состояние), регистр «0» является регистром приемопередатчика, а регистр «1» задает маску прерываний. Когда этот бит равен 1, то регистры «0» и «1» применяются для чтения/записи старшего и младшего байтов делителя, определяющего скорость обмена. Значение делителя определяется по формуле: $K = 115200/V$, где V – скорость передачи, бит/с.

Входная частота синхронизации 1,8432 МГц делится адаптером на K , и получается 16-кратная частота передачи.

Базовый адрес COM-порта определяется путем чтения переменных BIOS, адреса которых приведены в табл. 5.4. Стандартным адресом порта COM является 3F8h, COM2 – 2F8h. Перед работой с портом рекомендуется определить его адрес путем чтения переменной BIOS, а не брать стандартное значение. Назначение регистров UART и их адреса относительно базового приведены в табл. 5.5.

Таблица 5.4

Адреса переменных BIOS для COM портов

Имя порта	Адрес в BIOS	Тип переменной	Описание
COM1	0040:0000h	Word	Базовый адрес порта COM1. Если переменная равна 0, то порт не найден
COM2	0040:0002h	»»»	То же
COM3	0040:0004h	»»»	»»»
COM4	0040:0006h	»»»	»»»

Таблица 5.5

Регистры UART 16550A

Смещение	DLAB	R/W	8250	Значение
0	1	R/W	+-	Регистр скорости обмена (LO)
	0	WO	++	Регистр передатчика
	0	RO	++	Регистр приемника
1	1	R/W	+-	Регистр скорости обмена (HI)
	0	R/W	+-	Регистр маски разрешения прерываний
2	x	RO	+-	Регистр идентификации прерываний
	x	WO	-	Регистр управления FIFO
3	x	R/W	+	Регистр управления линией (настройка параметров канала)
	x			
4	x	R/W*	+	Регистр управления модемом
5	x	R/W*	+	Регистр состояния линии
6	x	R/W	+	Регистр состояния модема
7	x	R/W	-	Рабочий регистр для временного хранения данных

- Примечания.* 1. * – некоторые биты допускают только чтение.
 2. Совместимость с 8250: '+' – полная, '+-' – частичная, '-' – отсутствует в 8250.
 3. DLAB – состояние бита 7 регистра управления линией.
 4. При помещении байта в буфер передатчика он немедленно передается.

Описание регистров УАПП 16550А

В табл. 5.6 – 5.11 значком * отмечены биты, не используемые в 8250.

Регистр разрешения прерываний. Единичное значение бита соответствует разрешению прерывания, назначение битов приведено в табл. 5.6. Если работа с портом идет без использования прерываний, то их следует запретить путем записи 0 в этот регистр.

Таблица 5.6

Регистр разрешения прерываний

Бит	Значение
7,6,5,4	= 0 – не используются
3	По изменении статуса модема (STS, DSR, RI, RLSD)
2	По ошибке приема данных
1	По завершению передачи
0	По приему символа (в режиме FIFO – по тайм-ауту)

Регистр идентификации прерываний (табл. 5.7). Ввиду того, что контроллер обслуживается одним прерыванием, а возникать оно может, в общем случае, в 4 различных ситуациях, процедуре обработки прерывания необходимо выяснить, по какой причине произошло прерывание.

Таблица 5.7

Регистр идентификации прерываний

Бит	Значение
7,6*	Признак режима FIFO: 11 – FIFO 16550А, 10 – FIFO 16550; 00 – обычный
5, 4	Не используются
3*	Прерывание по тайм-ауту (не в режиме FIFO)
2,1	00 – прерывание от приемника; 01 – есть данные в приемнике; 10 – буфер передатчика пуст; 11 – изменение статуса модема
0	1 – нет запроса прерывания, 0 – есть запрос

Регистр управления FIFO (только для записи, только 16550+) (табл. 5.8). Для очистки буферов FIFO необходимо запретить и затем снова разрешить режим FIFO.

Регистр управления линией применяется для задания основных параметров адаптера (табл. 5.9). Бит 7 регистра управления линией называется DLAB и предназначен для определения назначения регистров «0» и «1». Он устанавливается в 1 только на время программирования делителя управления скоростью передачи.

Таблица 5.8

Регистр управления FIFO

Бит	Значение
7,6	Уровень заполнения FIFO-буфера, при котором генерируется прерывание (00 – 1 байт; 01 – 4 байт; 10 – 8 байт; 11 – 14 байт)
5,4	Резерв

Таблица 5.9

Регистр управления линией

Бит	Значение
7	DLAB. Управление доступом к регистрам адаптера интерфейса. 1 – режим программирования делителя, 0 – чтение/запись
6	Формирование "обрыва линии" (вывод нулей)
5	Принудительная четность (если 0, то бит паритета генерируется в соответствии с паритетом символа, 1 – всегда равен инверсному значению бита 4 этого регистра)
3	Разрешение операций DMA
2	Сброс счетчика FIFO передатчика
1	Сброс счетчика FIFO приемника
0	Разрешение (1) режима FIFO для передатчика и приемника. При смене режима FIFO буфера очищаются!

Бит 4 управляет типом паритета, бит 3 задает разрешение контроля паритета. Бит 2 задает длину стоп-бита. Биты 1-0 определяют длину слова. Наиболее часто применяются слова длиной 7 бит и 8 бит.

Регистр управления модемом позволяет управлять модемом, значение битов приведены в табл. 5.10.

Таблица 5.10

Регистр управления модемом

Бит	Значение
7,6,5	0 – резерв
4	0 – нормальный режим, 1 – режим диагностики
3	OUT2 инверсный (разрешить прерывания от RS-232)
2	OUT1 инверсный (используется только в режиме диагностики)
1	RTSC – управление выходом RTS (1 – активен (-V), 0 – пассивен (+V))
0	DTRC – управление выходом DTR (1 – активен (-V), 0 – пассивен (+V))

В режиме диагностики выход передатчика переводится в состояние "1", вход приемника отключается, выход сдвигающего регистра передатчика логически соединяется с входом приемника; входы DSR, CTS, RI, DCD отключаются от входных линий и соединяются с DTRC, RTSC, OUT1, OUT2 соответственно, выходы управления модемом переводятся в пассивное состояние. Переданный в таком режиме байт должен немедленно приниматься, что позволяет проверять внутренние каналы данных порта и сдвиговые регистры.

Регистр состояния линии позволяет определить состояние приемо-передатчика (табл. 5.11.).

Таблица 5.11

Регистр состояния линии

Бит	Значение
7*	Ошибка принятых данных в режиме FIFO (буфер FIFO содержит хотя бы один символ, принятый с ошибкой)
6	Буфер передатчика пуст (символов нет ни в регистре передатчика, ни в сдвиговом регистре, ни в буфере FIFO)
5	Сдвиговый регистр передатчика пуст (буфер FIFO передатчика пуст). Может генерировать прерывание.
4	BD – индикатор обрыва линии (на входе приемника 0 не менее, чем время посылки одного символа)

Окончание табл. 5.11

Бит	Значение
3	FE – ошибка кадра (стоп-бит не найден)
2	PE – ошибка контрольного бита
1	OE – ошибка переполнения. Означает потерю символа (имеет аналогичный смысл в режиме FIFO)
0	DR – принятые данные готовы. Сбрасывается при чтении регистра приемника

Регистр состояния модема (табл. 5.12) позволяет определить состояние модема. Биты 3-0 вызывают прерывание "Изменение состояние модема". "1" в любом из этих битов свидетельствует о том, что состояние линии изменилось по сравнению с состоянием на момент последнего чтения этого регистра.

Таблица 5.12

Регистр состояния модема

Бит	Значение
7	Состояние линии DSD
6	» » RI
5	» » DSR
4	» » CTS
3	Изменение DSD
2	» RI
1	» DSR
0	» STS

Примеры работы с последовательным портом на низком уровне

Работа с последовательным портом на низком уровне аналогична работе с параллельным: вначале определяем базовый адрес порта, затем работаем с его регистрами. Главная отличительная особенность состоит в необходимости установки параметров COM-порта перед началом работы. Приведем пример программы настройки параметров порта (скорость 4800 кбод, 1 стоп-бит, четность, 8 бит/слово):

```
mov ax, 40h
mov es, ax; в es - сегмент = 0040h
mov dx, es:[00] ; dx = базовый адрес порта COM1
```

```

    mov COM1_adr, dx ; запомнили адрес порта COM1 в
переменной
    add dx, 3 ; DX=адресу регистра управления
    mov al, 80h ; Установили бит DLAB - настройка де-
делителя
    out dx, al ; Рабочая скорость - 4800 Кбод
    dec dx
    dec dx ; DX=адресу старшего байта делителя скорости
    mov al, 0
    out dx, al
    dec dx ; DX=адресу младшего байта делителя скоро-
сти
    mov al, 18h ; Установка младшего байта делителя
    out dx, al
    add dx, 3 ; DX=адресу регистра управления
    mov al, 00011011b ; DLAB=0, четность, 1 стоп-бит,
8 бит/слово
    out dx, al
    dec dx
    dec dx ; DX=адресу регистра разрешения прерываний
    mov al, 00h ; Прерывания запрещены
    out dx, al

```

Пример чтения байта с ожиданием без использования прерываний:

```

mov dx, COM1_adr ; DX=базовому адресу COM1
add dx, 5 ; DX=адресу регистра состояния
Wait:
in al, dx ; чтение байта состояния
test al, 01h ; Бит 1="1" (принят байт)
jz wait ; Нет - ждем
sub dx, 5 ; DX=адресу регистра приемника
in al, dx ; чтение принятого байта

```

Пример передачи байта из АН (без контроля готовности приемника):

```

mov dx, COM1_adr ; DX=базовому адресу COM1
add dx, 5 ; DX=адресу регистра состояния
Wait:
in al, dx ; чтение байта состояния

```

```

test al, 40h ; Бит 6=1 (готов к передаче очеред-
ного байта)
jz wait ; Нет - ждем
sub dx, 5 ; DX=адресу регистра передатчика
mov al, ah
out dx, al ; Передача

```

5.2. Интерфейс RS-485

Интерфейс RS-485 (другое название EIA/TIA-485) – один из наиболее распространенных стандартов физического уровня связи. Физический уровень – это канал связи и способ передачи сигнала (1-й уровень модели взаимосвязи открытых систем OSI).

Сеть, построенная на интерфейсе RS-485, представляет собой приемопередатчики, соединенные при помощи витой пары – двух скрученных проводов. В основе интерфейса RS-485 лежит принцип дифференциальной (балансной) передачи данных. Суть его заключается в передаче одного сигнала по двум проводам. Причем по одному проводу (условно линия *A*) идет оригинальный сигнал, а по другому (условно линия *B*) – его инверсная копия. Другими словами, если на одном проводе "1", то на другом "0", и наоборот. Таким образом, между двумя проводами витой пары всегда есть разность потенциалов: при "1" она положительна, при "0" – отрицательна (рис. 5.6).

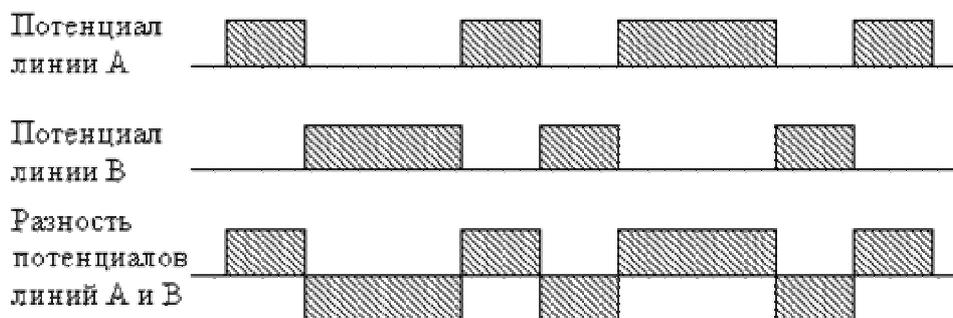


Рис. 5.6. Сигнал в линии

Именно этой разностью потенциалов и передается сигнал. Такой способ передачи обеспечивает высокую устойчивость к синфазной помехе. Синфазной называют помеху, действующую на оба провода линии одинаково. К примеру, электромагнитная волна, проходя через участок линии связи, наводит в обоих проводах потенциал. Если сигнал передается по-

тенциалом в одном проводе относительно общего, как в RS-232, то наводка на этот провод может исказить сигнал относительно хорошо поглощающего наводки общего ("земли"). Кроме того, на сопротивлении длинного общего провода будет падать разность потенциалов земель – дополнительный источник искажений. А при дифференциальной передаче искажения не происходит. В самом деле, если два провода пролегают близко друг к другу, да еще перевиты, то наводка на оба провода одинакова. Потенциал в обоих одинаково нагруженных проводах изменяется одинаково, при этом информативная разность потенциалов остается без изменений.

Аппаратная реализация интерфейса – микросхемы приемопередатчиков с дифференциальными входами/выходами (к линии) и цифровыми портами (к портам UART-контроллера). Существуют два варианта такого интерфейса: RS-422 и RS-485 (рис. 5.7).

RS-422 – полнодуплексный интерфейс. Прием и передача идут по двум отдельным парам проводов. На каждой паре проводов может быть только по одному передатчику.

RS-485 – полудуплексный интерфейс. Прием и передача идут по одной паре проводов с разделением по времени. В сети может быть много передатчиков, так как они отключаются в режиме приема.

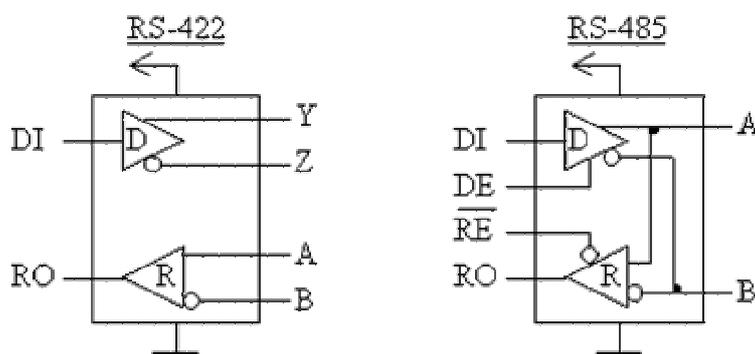


Рис. 5.7. Приемопередатчики интерфейсов:
D (driver) – передатчик; R (receiver) – приемник;
DI (driver input) – цифровой вход передатчика;
RO (receiver output) – цифровой выход приемника;
DE (driver enable) – разрешение работы передатчика;
RE (receiver enable) – разрешение работы приемника; A – прямой дифференциальный вход/выход;
B – инверсный дифференциальный вход/выход;
Y – прямой дифференциальный выход (RS-422);
Z – инверсный дифференциальный выход (RS-422).

Остановимся поподробнее на приемопередатчике RS-485 (рис. 5.8). Цифровой выход приемника (RO) подключается к порту приемника UART

(RX), цифровой вход передатчика (DI) – к порту передатчика UART (TX). Поскольку на дифференциальной стороне приемник и передатчик соединены, то во время приема нужно отключать передатчик, а во время передачи – приемник. Для этого служат управляющие входы – разрешение приемника (RE) и разрешение передатчика (DE). Так как вход RE инверсный, то его можно соединить с DE и переключать приемник и передатчик одним сигналом с любого порта контроллера. При уровне "0" – работа на прием, при "1" – на передачу.

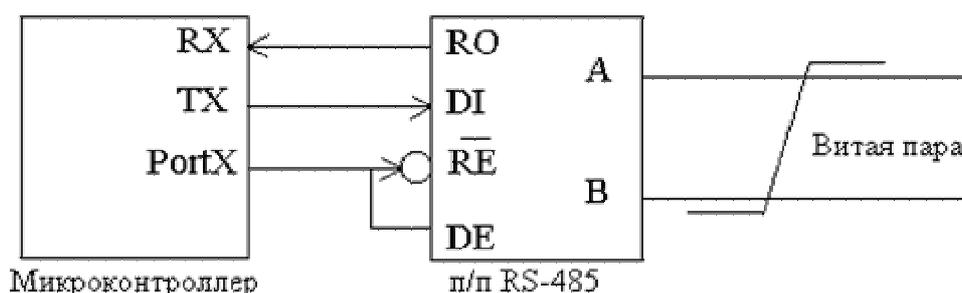


Рис. 5.8. Принцип работы приемопередатчика

Приемник, получая на дифференциальных входах (AB) разность потенциалов (U_{AB}), переводит их в цифровой сигнал на выходе RO. Чувствительность приемника может быть разной, но гарантированный пороговый диапазон распознавания сигнала производители микросхем приемопередатчиков пишут в документации (табл. 5.13). Обычно эти пороги составляют ± 200 мВ. То есть, когда $U_{AB} > +200$ мВ – приемник определяет "1", когда $U_{AB} < -200$ мВ – приемник определяет "0". Если разность потенциалов в линии настолько мала, что не выходит за пороговые значения, – правильное распознавание сигнала не гарантируется. Кроме того, в линии могут быть и несинфазные помехи, которые исказят столь слабый сигнал.

Все устройства подключаются к одной витой паре одинаково: прямые выходы (A) – к одному проводу, инверсные (B) – к другому.

Входное сопротивление приемника со стороны линии (R_{AB}) обычно составляет 12 кОм. Так как мощность передатчика не беспредельна, это создает ограничение на количество приемников, подключенных к линии. Согласно спецификации RS-485 с учетом согласующих резисторов передатчик может вести до 32 приемников. Однако есть ряд микросхем с по-

вышенным входным сопротивлением, что позволяет подключить к линии значительно больше 32 устройств.

Максимальная скорость связи по спецификации RS-485 может достигать 10 Мбод/с. Максимальное расстояние – 1200 м. Если необходимо организовать связь на расстоянии большем 1200 м или подключить больше устройств, чем допускает нагрузочная способность передатчика, – применяют специальные повторители (репитеры).

Таблица 5.13

Стандартные параметры интерфейсов

Параметр	RS-422	RS-485
Допустимое число передатчиков / приемников	1 / 10	32 / 32
Максимальная длина кабеля, м	1200	1200
Максимальная скорость связи, Мбит/с	10	10
Диапазон напряжений "1" передатчика, В	+2...+10	+1,5...+6
Диапазон напряжений "0" передатчика, В	-2...-10	-1.5...-6
Диапазон синфазного напряжения передатчика, В	-3...+3	-1...+3
Допустимый диапазон напряжений приемника, В	-7...+7	-7...+12
Пороговый диапазон чувствительности приемника, мВ	±200	±200
Максимальный ток короткого замыкания драйвера, мА	150	250
Допустимое сопротивление нагрузки передатчика, Ом	100	54
Входное сопротивление приемника, кОм	4	12
Максимальное время нарастания сигнала передатчика, %	10	30

5.3. Согласование и конфигурация линии связи

При больших расстояниях между устройствами, связанными по витой паре, и высоких скоростях передачи начинают проявляться так называемые эффекты длинных линий. Причина этому – конечность скорости распространения электромагнитных волн в проводниках. Скорость эта существенно меньше скорости света в вакууме и составляет немногим боль-

ше 200 мм/нс. Электрический сигнал имеет также свойство отражаться от открытых концов линии передачи и ее ответвлений. Грубая аналогия – желоб, наполненный водой. Волна, созданная в одном конце, идет по желобу и, отразившись от стенки в конце, идет обратно, отражается опять и так далее, пока не затухнет. Для коротких линий и малых скоростей передачи этот процесс происходит так быстро, что остается незамеченным. Однако, время реакции приемников – десятки-сотни наносекунд. В таком масштабе времени несколько десятков метров электрический сигнал проходит отнюдь не мгновенно. И если расстояние достаточно большое, фронт сигнала, отразившийся в конце линии и вернувшийся обратно, может исказить текущий или следующий сигнал. В таких случаях нужно каким-то образом подавлять эффект отражения.

У любой линии связи есть такой параметр, как волновое сопротивление Z_v . Оно зависит от характеристик используемого кабеля, но не от его длины. Для обычно применяемых в линиях связи витых пар $Z_v = 120$ Ом. Оказывается, что если на удаленном конце линии между проводниками витой пары включить резистор с номиналом, равным волновому сопротивлению линии, то электромагнитная волна, дошедшая до "тупика", поглощается на таком резисторе. Отсюда его название "согласующий резистор" или "терминатор".

Большой минус согласования на резисторах – повышенное потребление тока от передатчика, ведь в линию включается низкоомная нагрузка. Поэтому рекомендуется включать передатчик только на время отправки посылки. Есть способы уменьшить потребление тока, включая последовательно с согласующим резистором конденсатор для развязки по постоянному току. Однако такой способ имеет свои недостатки. Для коротких линий (несколько десятков метров) и низких скоростей (меньше 38400 бод) согласование можно вообще не делать.

Эффект отражения и необходимость правильного согласования накладывают ограничения на конфигурацию линии связи.

Линия связи должна представлять собой один кабель витой пары. К этому кабелю присоединяются все приемники и передатчики. Расстояние от линии до микросхем интерфейса RS-485 должно быть как можно короче, так как длинные ответвления вносят рассогласование и вызывают отражения.

В оба наиболее удаленных конца кабеля ($Z_v = 120$ Ом) включают согласующие резисторы R_t по 120 Ом (0,25 Вт). Если в системе только один

передатчик и он находится в конце линии, то достаточно одного согласующего резистора на противоположном конце линии (рис. 5.9).

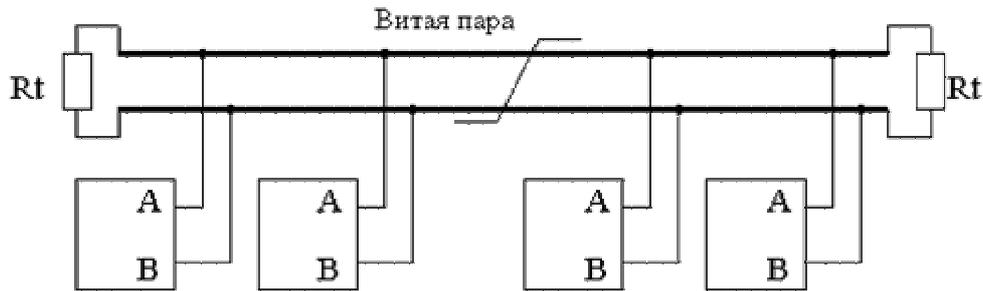


Рис. 5.9. Согласование линии

5.4. Защитное смещение

Как уже упоминалось, приемники большинства микросхем RS-485 имеют пороговый диапазон распознавания сигнала на входах А-В – $\pm 200\text{ мВ}$. Если $|U_{AB}|$ меньше порогового (около 0), то на выходе приемника RO могут быть произвольные логические уровни из-за несинфазной помехи. Такое может случиться либо при отсоединении приемника от линии, либо при отсутствии в линии активных передатчиков, когда никто не задает уровень. Чтобы в этих ситуациях избежать выдачи ошибочных сигналов на приемник UART, необходимо на входах А-В гарантировать разность потенциалов $U_{AB} > +200\text{ мВ}$. Это смещение при отсутствии входных сигналов обеспечивает на выходе приемника логическую "1", поддерживая, таким образом, уровень стопового бита.

Добиться этого просто - прямой вход (А) следует подтянуть к питанию, а инверсный (В) – к "земле". Получается делитель (рис. 5.10).

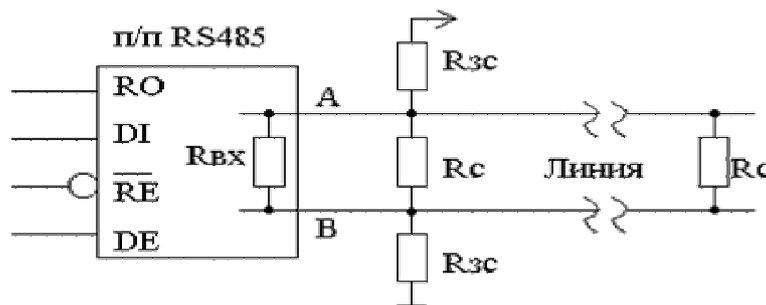


Рис. 5.10. Организация делителя напряжения:

$R_{вх}$ – входное сопротивление приемника (обычно 12 кОм);

R_c – согласующие резисторы (120 Ом);

$R_{зс}$ – резисторы защитного смещения

Величины сопротивлений для резисторов защитного смещения ($R_{зс}$) нетрудно рассчитать по делителю. Необходимо обеспечить $U_{AB} > 200\text{ мВ}$.

Напряжение питания – 5 В. Сопротивление среднего плеча – 120 Ом//120 Ом//12 кОм на каждый приемник – примерно 57 Ом (для 10 приемников). Таким образом, выходит примерно по 650 Ом на каждый из двух $R_{з.с}$. Для смещения с запасом сопротивление $R_{з.с}$ должно быть меньше 650 Ом. Традиционно ставят 560 Ом.

Обратите внимание: в расчете номинала $R_{з.с}$ учитывается нагрузка. Если на линии висит много приемников, то номинал $R_{з.с}$ должен быть меньше. В длинных линиях передачи необходимо также учитывать сопротивление витой пары, которое может "съесть" часть смещающей разности потенциалов для удаленных от места подтяжки устройств. Для длинной линии лучше ставить два комплекта подтягивающих резисторов в оба удаленных конца рядом с терминаторами.

Многие производители приемопередатчиков заявляют о функции безотказности (failsafe) своих изделий, заключающейся во встроенном смещении. Следует различать два вида такой защиты.

Безотказность в открытых цепях (Open circuit failsafe). В таких приемопередатчиках применяются встроенные подтягивающие резисторы. Эти резисторы, как правило, высокоомные, чтобы уменьшить потребление тока. Из-за этого необходимое смещение обеспечивается только для открытых (ненагруженных) дифференциальных входов. В самом деле, если приемник отключен от линии или она не нагружена, тогда в среднем плече делителя остается только большое входное сопротивление, на котором и падает необходимая разность потенциалов. Однако если приемопередатчик нагрузить на линию с двумя согласующими резисторами по 120 Ом, то в среднем плече делителя оказывается меньше 60 Ом, на которых, по сравнению с высокоомными подтяжками, ничего существенного не падает. Поэтому если в нагруженной линии нет активных передатчиков, то встроенные резисторы не обеспечивают достаточное смещение. В этом случае остается необходимость устанавливать внешние резисторы защитного смещения, как это было описано выше.

Истинная безотказность (True failsafe.) В этих устройствах смещены сами пороги распознавания сигнала. Например: -50 / -200 мВ вместо стандартных порогов ± 200 мВ. То есть при $U_{AB} > -50$ мВ на выходе приемника RO будет логическая "1", а при $U_{AB} < -200$ – на RO будет "0". Таким образом, и в разомкнутой и в пассивной линии при разности потенциалов U_{AB} , близкой к нулю, приемник выдаст "1". Для таких приемопередатчиков внешнее защитное смещение не требуется. Тем не менее, для лучшей помехозащищенности все-таки стоит дополнительно немного подтягивать линию.

Сразу виден минус внешнего защитного смещения: через делитель постоянно будет протекать ток, что может быть недопустимо в системах малого потребления. В таком случае можно сделать следующее:

1. Уменьшить потребление тока, увеличив сопротивления $R_{з.с}$. Хотя производители приемопередатчиков и пишут о пороге распознавания в 200 мВ, на практике вполне хватает 100 мВ и даже меньше. Таким образом, можно сразу увеличить сопротивления $R_{з.с}$ раза в два-три. Помехозащищенность при этом несколько снижается, но во многих случаях это не критично.

2. Использовать true failsafe приемопередатчики со смещенными порогами распознавания. Например, у микросхем MAX3080 и MAX3471 пороги: -50 мВ / -200 мВ, что гарантирует единичный уровень на выходе приемника при отсутствии смещения ($U_{AB} = 0$). Тогда внешние резисторы защитного смещения можно убрать или значительно увеличить их сопротивление.

3. Не применять без необходимости согласование на резисторах. Если линия не будет нагружена на 2 резистора по 120 Ом, то для обеспечения защитного смещения хватит подтяжек в несколько килоом в зависимости от числа приемников на линии.

Для оптически развязанной линии подтягивать потенциал следует к питанию и "земле" изолированной линии. Если не применяется опторазвязка, подтягивать можно к любому питанию, так как делитель создаст лишь небольшую разность потенциалов между линиями А и В. Нужно только помнить о возможной разности потенциалов между "землями" устройств, расположенных далеко друг от друга.

5.5. Исключение приема при передаче в полудуплексном режиме

При работе с полудуплексным интерфейсом RS-485 (прием и передача по одной паре проводов с разделением по времени) можно забыть, что UART контроллер – полнодуплексный, то есть принимает и передает сигнал независимо и одновременно.

Обычно во время работы приемопередатчика RS-485 на передачу, выход приемника RO переводится в третье состояние и ножка RX контроллера (приемник UART) "повисает в воздухе" (рис. 5.11). В результате во время передачи на приемнике UART вместо уровня стопового бита ("1")

окажется неизвестно что и любая помеха будет принята за входной сигнал. Поэтому нужно либо на время передачи отключать приемник UART (через управляющий регистр), либо подтягивать RX к единице. У некоторых микроконтроллеров это можно сделать программно – активировать встроенные подтяжки портов.

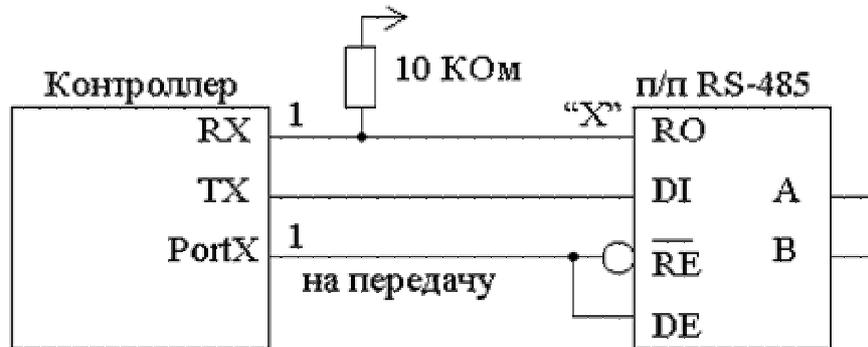


Рис. 5.11. Схема исключения приема при передаче

5.6. "Горячее" подключение к линии связи

Насколько известно, спецификацией RS-485 не предусмотрено "горячее" подключение – включение новых приемопередатчиков в линию связи во время работы системы. Тем не менее, подобную операцию система переносит практически безболезненно, если учесть один нюанс. Это важно, когда питание на устройство подается в момент подключения, например, когда плата в виде кассеты вставляется в разъем. Дело в том, что во время любого сброса: по включению питания, по сигналу на входе "Reset", по срабатыванию сторожевого таймера – контроллеру требуется время на инициализацию, которое может составлять до нескольких десятков миллисекунд. Пока контроллер находится в состоянии сброса, он принудительно настраивает все порты на вход. Получается ситуация, при которой питание на микросхему приемопередатчика RS-485 уже подано, но входы разрешения приемника RE и передатчика DE "висят в воздухе". В результате приемопередатчик может открыться по помехе на передачу и все время, пока микроконтроллер отключен, пропускать в работающую линию шум. Избежать этого легко: достаточно через резистор в несколько килоом подтянуть вход разрешения приемника RE к нулю (рис. 5.12). Этим приемопе-

передатчик сразу по включении питания настраивается на прием и не обращается к линии.

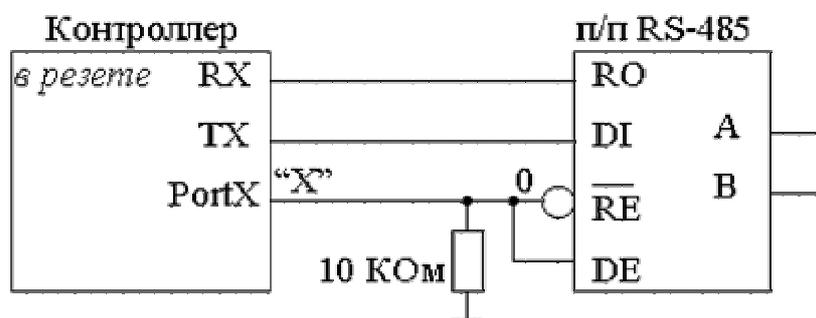


Рис. 5.12. Схема защиты от помех

5.7. Рекомендации по организации протокола связи

На физическом уровне линия связи готова к работе, однако нужен еще и протокол – договоренность между устройствами системы о формате посылок.

По принципу работы интерфейса RS-485 устройства не могут передавать сигналы одновременно – будет конфликт передатчиков. Следовательно, требуется распределить между устройствами право на передачу. Отсюда основное деление: централизованный (одномастерный) и децентрализованный (многомастерный) обмены.

В централизованной сети одно устройство всегда ведущее (мастер). Оно генерирует запросы и команды остальным (ведомым) устройствам. Ведомые устройства могут передавать только по команде ведущего. Как правило, обмен между ведомыми идет только через ведущего, хотя для ускорения обмена можно организовать передачу данных от одного ведомого к другому по команде ведущего.

В децентрализованной сети роль ведущего может передаваться от устройству к устройству либо по некоторому алгоритму очередности, либо по команде текущего ведущего к следующему (передача маркера ведущего). При этом ведомое устройство может в своем ответе ведущему передать запрос на переход в режим ведущего и ожидать разрешения или запрета.

Последовательный канал по сравнению с контроллером работает медленно. На скорости 9600 бод передача одного символа занимает больше миллисекунды. Поэтому, когда контроллер плотно загружен вычислениями и не должен их останавливать на время обмена по UART, нужно использовать прерывания по завершению приема и передачи символа. Можно выделить место в памяти для формирования посылки на передачу и сохранения принятой посылки (буфер посылки), а также указатели на позицию текущего символа. Прерывания по завершению приема или передачи символа вызывают соответствующие подпрограммы, которые передают или сохраняют очередной символ со сдвигом указателя и проверкой признака конца сообщения, после чего возвращают управление основной программе до следующего прерывания. По завершению отправки или приема всей посылки либо формируется пользовательский флаг, отрабатываемый в основном цикле программы, либо сразу вызывается подпрограмма обработки сообщения.

В общем случае посылка по последовательному каналу состоит из управляющих байтов (синхронизация посылки, адресов отправителя и получателя, контрольной суммы и прочее) и собственно байтов данных.

Протоколов существует множество, и можно придумать еще больше, но лучше пользоваться наиболее употребительными из них. Одним из стандартных протоколов последовательной передачи является MODBUS, его поддержку обеспечивают многие производители промышленных контроллеров.

Основная задача в организации протокола – заставить все устройства различать управляющие байты и байты данных. К примеру, ведомое устройство, получая по линии поток байтов, должно понимать, где начало посылки, где конец и кому она адресована.

Часто встречаются протоколы на основе ASCII-кода. Управляющие символы и данные передаются в виде обыкновенных ASCII-символов. Посылка может выглядеть так:

В HEX : 3Ah 31h 32h 52h 53h 34h 38h 35h 0Dh

В ASCII : ":" "1" "2" "R" "S" "4" "8" "5" /ПС/

В начале – управляющий символ начала посылки ":", следующие две цифры - адрес получателя (12), затем символы данных (RS-485) и в конце - управляющий символ конца посылки 0Dh (перевод строки). Все устройст-

ва на линии, приняв символ ":", начинают записывать в память посылку до символа конца строки 0Dh. Затем сравнивают адрес из посылки со своим адресом. Устройство с совпавшим адресом обрабатывает данные посылки, остальные – игнорируют посылку. Данные могут содержать любые символы, кроме управляющих (":", 0Dh).

Достоинство этого протокола в удобстве отладки системы и простоте синхронизации посылок. Можно через преобразователь RS-485 – RS-232 подключить линию к СОМ-порту компьютера и в любом терминальном устройстве увидеть всю проходящую информацию "на человеческом языке". Недостатки – относительно большой размер посылки при передаче большого количества двоичной информации, ведь на передачу каждого байта нужно два ASCII-символа (7Fh – "7", "F"). Кроме того, надо преобразовывать данные из двоичного вида в ASCII и обратно.

Можно организовать протокол с непосредственной передачей двоичных данных. При этом управляющие символы и байты данных различаются с помощью настройки дополнительного девятого бита в UART. Для управляющих символов этот бит устанавливается в "1". Первым в посылке передается управляющий символ с единичным девятым битом, остальные его "нормальные" биты могут содержать адрес устройства-получателя, признак начала/конца посылки и что-нибудь еще. Затем передаются байты данных с нулевым девятым битом. Все принимающие устройства узнают по девятому биту управляющий символ и по содержанию его остальных битов определяют, кому адресованы последующие данные. Адресуемое устройство принимает данные, а все остальные игнорируют их до следующего управляющего символа.

UART некоторых контроллеров, например C167 (Infineon), может в особом режиме (wakeur) автоматически распознавать в полученном байте девятый бит и генерировать прерывание при получении только управляющего символа. Адресуемое устройство при этом нужно переключить в режим обычного приема до следующего управляющего символа. Это позволяет остальным устройствам экономить время на обработке прерываний при получении байтов данных, адресованных не им.

Если требуется сопряжение системы и компьютера с Windows, такой протокол лучше не применять, так как у Windows могут быть проблемы с распознаванием девятого бита в UART.

Протокол может быть "чисто" двоичным, то есть без выделения специальных управляющих символов. Синхронизация посылок в этом случае может осуществляться за счет отслеживания паузы между принятыми байтами. Принимающее устройство отсчитывает время с момента последнего приема байта до следующего, и если эта пауза оказывается больше какой-то величины (например, 1,5 – 3,5 байта), делается вывод о потере предыдущей посылки и начале новой. Даже если предыдущая посылка была не закончена – приемный буфер сбрасывается. Можно также синхронизировать посылки по уникальной стартовой последовательности байтов (по аналогии со стартовым символом в ASCII-протоколе). В таких протоколах надо принимать особые меры для защиты от приема ложной посылки, начатой из-за помехи.

5.8. Программные методы борьбы со сбоями

Для повышения надежности связи обязательно нужно предусмотреть программные методы борьбы со сбоями. Их можно условно разделить на две группы: защита от рассинхронизации и контроль достоверности.

Защита от рассинхронизации

Несмотря на защитное смещение, сильная помеха может пробиться в линию без активных передатчиков и нарушить правильную последовательность приема посылок. Тогда возникает необходимость первой же нормальной посылкой вразумить принимающие устройства и не дать им принять помеху за посылку. Делается это с помощью синхронизации кадров (активная пауза) и синхронизации посылок (преамбула).

Защита от рассинхронизации кадров (рис. 5.13). *Обязательная мера! Все последующие меры синхронизации посылок имеют смысл только совместно с этой.* Помеха ложным старт-битом может сбить правильный прием кадров последующей посылки. Чтобы вернуться к верной последовательности, нужно сделать паузу между включением приемопередатчика на передачу и посылкой данных. Все это время передатчик удерживает в линии высокий уровень, через который помехе трудно пробиться (активная пауза). Паузы длительностью в 1 кадр на данной скорости связи (10 – 11 бит) будет достаточно для того, чтобы любое устройство, принимавшее

помехи, приняло стоп-бит. Тогда следующий кадр будет приниматься с нормального старт-бита.



Рис. 5.13. Диаграмма защиты от рассинхронизации кадров

Того же эффекта можно добиться передачей символа FFh перед первым байтом посылки, так как кроме старт-бита, все его биты – "1". (Если старт-бит символа FFh попадет на стоп-бит ложного кадра, будет просто засчитана ошибка кадра.)

Защита от рассинхронизации посылок. Применяется совместно с предыдущей защитой! Особо вредная помеха может замаскироваться под управляющий символ и сбить принимаемую затем посылку. Кроме того, предыдущая посылка может быть прервана. Из-за этого крайне важно в подпрограмме приема и сохранения данных предусмотреть меры по опознанию настоящего начала посылки и сбросу приемного буфера посылки (области памяти, куда сохраняются принимаемые байты). Для этого служит преамбула – предварительный признак начала посылки.

Стартовый символ. В ASCII-протоколе роль преамбулы играет специальный управляющий символ начала посылки. По каждому приему такого символа нужно сбрасывать буфер: обнулять число принятых байт, перемещать указатель на начало буфера и т.п. То же самое нужно делать при переполнении буфера. Это позволит настоящему управляющему символу сбросить предыдущую "посылку", начатую ложным символом.

Пример. Последний управляющий символ ":" сбросит предыдущую ложную посылку:

____ :) ____ : 1 2 R S 4 8 5 /ПС/ ____

Стартовая пауза. В двоичном протоколе, где не предусмотрен уникальный управляющий символ и синхронизация посылок идет по заданной паузе между байтами, достаточно увеличить активную паузу, описанную в синхронизации кадров, до длительности паузы между байтами, по которой начинается прием новой посылки. То есть между включением приемопе-

передатчика на передачу и отправкой первых байтов посылки нужно сделать паузу длительностью в 1,5 – 3,5 кадра UART. При активном передатчике во время такой преамбулы помехе трудно будет прорваться к приемникам, они зафиксируют нужную паузу, сбросят буфер посылки и настроятся на прием новой посылки. Этот метод применяется, в частности, для протокола MODBUS RTU.

Стартовая последовательность. Если в двоичном протоколе синхронизация осуществляется лишь по корректному началу посылки, то отфильтровать ложную посылку можно только по логике ее структуры. Преамбула в данном случае – некоторая стартовая последовательность символов, которая не может встретиться в данных посылки, и которую вряд ли сформирует помеха. Преамбула отсылается перед основной посылкой. Принимающее устройство отслеживает в поступающих данных эту стартовую последовательность. Где бы она не состоялась, принимающее устройство сбрасывает буфер посылки и начинает принимать новую.

Вариант 1. Посылка начинает заново приниматься после приема "go!" (вместо символов могут быть любые 8-битные данные):

_____ : - Ъ ___ g o ! 1 2 R S 4 8 5 _____

Вариант 2. Посылка начинает заново приниматься после приема *не менее* трех "E" подряд и стартового байта ":" (вместо символов могут быть любые 8-битные данные):

_____ > : - E ___ E E E : 1 2 R S 4 8 5 _____

Даже если до стартовой последовательности было два таких символа подряд, посылка начнет сохраняться только за последовательностью из не менее чем трех подряд (лишние игнорируются) и стартового символов. Если вместо "E" использовать байт FFh – можно совместить синхронизацию кадров и посылок. Для этого посылаются четыре FFh, а принимающее устройство ожидает не менее трех, с учетом того, что первый байт FFh может уйти на синхронизацию кадров.

Контроль достоверности

Особо сильная помеха может вклиниться в посылку, исказить управляющие символы или данные в ней, а то и вовсе уничтожить ее. Кроме того, одно из подключенных к линии устройств (абонент) может выйти из строя и перестать отвечать на запросы. На случай такой беды существуют контрольная сумма, тайм-ауты и квитирование.

Контрольная сумма – в общем случае 1 – 2 байта кода, полученного некоторым преобразованием из данных посылки. Самое простое – "исключающее или" всем байтам данных. Контрольная сумма рассчитывается и включается в посылку перед отправкой. Принимающее устройство производит ту же операцию над принятыми данными и сверяет рассчитанную контрольную сумму с полученной. Если посылка была повреждена, то, скорее всего, они не совпадут. В случае применения ASCII-протокола – код контрольной суммы также передается ASCII-символами.

Тайм-аут – максимальное время ожидания ответа от запрашиваемого устройства. Если посылка была повреждена или запрашиваемое устройство вышло из строя, то ведущее устройство не повиснет в ожидании ответа, а по истечении определенного времени признает наличие сбоя. После чего можно еще пару раз повторить запрос и, если сбой повторяется, перейти на отработку аварийной ситуации. Тайм-аут отсчитывается с момента завершения передачи запроса. Его длительность должна с небольшим запасом превышать максимальное время ответной передачи, плюс время, необходимое на обработку запроса и формирование ответа. Ведомому устройству тоже не мешает отработка тайм-аутов. Особенно в ситуациях, когда отсутствие регулярного обновления данных или новых команд от ведущего устройства критично для работы устройств системы. Самая простая реализация для ведомого – сброс сторожевого таймера по приему посылки. Если по какой-либо причине данные перестали поступать – устройство сбросится по переполнению сторожевого таймера. После сброса устанавливается безопасный режим до приема первой команды.

Квитирование – подтверждение доставки (квитанция). Когда важно, чтобы ведомый обязательно получил данные или команду, возникает необходимость проконтролировать получение им посылки. Ведущее устройство, отправив ведомому данные, ждет ответа с подтверждением. Ведомое устройство, получив данные, в случае их корректности посылает ответ, подтверждающий доставку. Если по истечении тайм-аута ведущее устройство не получает подтверждение, делается вывод о сбое в связи или в ведомом устройстве. Дальше обычные меры – повтор посылки. Но тут есть нюанс: повреждена и не получена может быть сама квитанция. Ведущее

устройство, не получив квитанцию, повторяет посылку, и ведомое обрабатывает ее повторно. Не всегда это существенно, но если перепосылалась команда типа "увеличить параметр на 1" это может привести к незапланированному двойному увеличению параметра. В таком случае надо предусмотреть что-нибудь типа циклической нумерации посылок, чтобы ведомое устройство отличало повторные посылки от новых и не обрабатывало их.

5.9. Защита устройств от перенапряжений в линии связи

Разность потенциалов между проводниками линии и между линией и "землей" приемопередатчика, как правило, не должна выходить за пределы -7...+12 В. Следовательно, может потребоваться защита от разности потенциалов между "землями" и от перенапряжений из-за замыкания на высоковольтные цепи.

Разность потенциалов между "землями". При организации сети на основе интерфейса RS-485 следует учитывать неявное присутствие третьего проводника – "земли". Ведь все приемопередатчики имеют питание и "землю". Если устройства расположены недалеко от начального источника питания, то разность потенциалов между "землями" устройств в сети невелика. Но если устройства находятся далеко друг от друга и получают местное питание, то между их "землями" может оказаться существенная разность потенциалов. Возможные последствия – выход из строя приемопередатчика, а то и всего устройства. В таких случаях следует применять гальваническую развязку или дренажный провод.

Гальваническая развязка линии и устройств осуществляется либо опторазвязкой цифровых сигналов (RO, DI, RE, DE) с организацией изолированного питания микросхем приемопередатчиков, либо применением приемопередатчиков со встроенной гальванической развязкой сигналов и питания (например, MAX1480). Тогда вместе с дифференциальными проводниками прокладываются провод изолированной "земли" (сигнальной "земли") и, возможно, провод изолированного питания линии.

Дренажный провод – провод, прокладываемый вместе с витой парой и соединяющий "земли" удаленных устройств (рис. 5.14). Через этот провод уравниваются потенциалы "земель". При включении устройства в ли-

нию дренажный провод следует подсоединять первым, а при отключении – отсоединять последним. Для ограничения тока через дренажный провод его заземляют в каждом устройстве через резистор в 100 Ом (0,5 Вт).

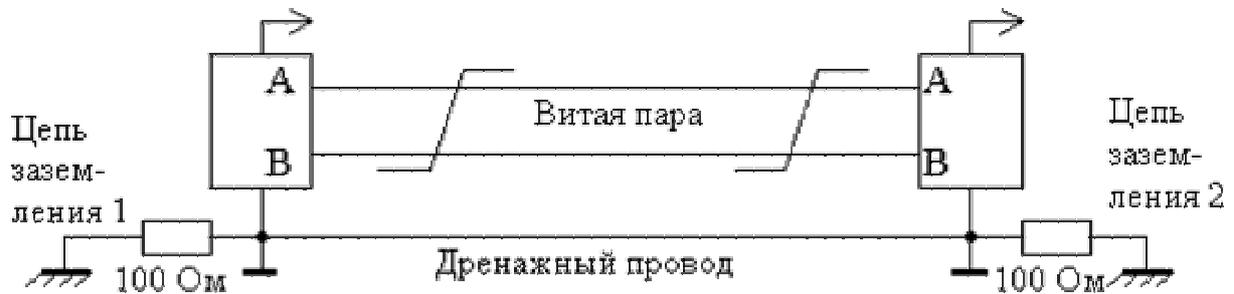


Рис. 5.14. Схема защиты приемопередатчиков

Замыкание на высоковольтные цепи. Если существует опасность попадания на линию или одну из местных "земель" высокого напряжения, следует применять опторазвязку или шунтирующие ограничители напряжения. А лучше и то и другое.

Напряжение пробоя опторазвязанного интерфейса составляет сотни и даже тысячи вольт. Это хорошо защищает устройство от перенапряжения, общего для всех проводников линии. Однако при дифференциальных перенапряжениях, когда высокий потенциал оказывается на одном из проводников, сам приемопередатчик будет поврежден.

Для защиты от дифференциальных перенапряжений все проводники линии, включая изолированный общий, шунтируются на локальные "земли" при помощи ограничителей напряжения. Это могут быть варисторы, полупроводниковые ограничители напряжения и газоразрядные трубки. Физический принцип их действия разный, но суть одна – при напряжении выше порогового их сопротивление резко падает, и они шунтируют линию. Газоразрядные трубки могут шунтировать очень большие токи, но имеют высокий порог пробоя и низкое быстродействие, поэтому их лучше применять по трехступенчатой схеме вместе с полупроводниковыми ограничителями. Когда заземление линии невозможно, проводники линии шунтируются ограничителями между собой (рис. 5.15). Но это защитит толь-

ко от дифференциальных перенапряжений – защиту от общего должна взять на себя опторазвязка.

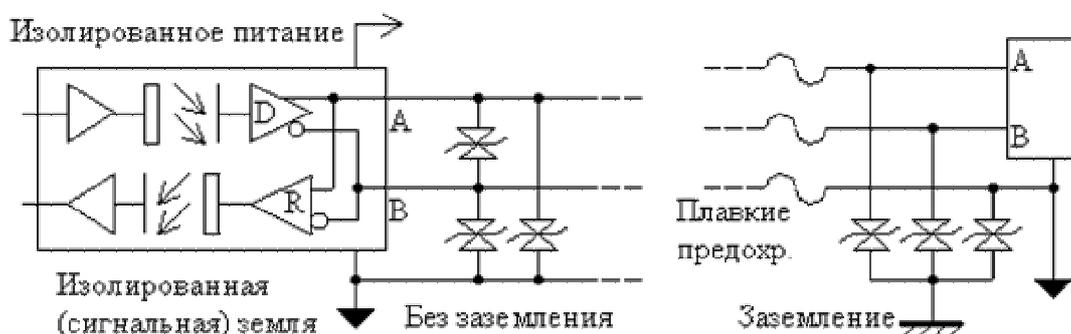


Рис. 5.15. Схемы защиты линии связи

Защита ограничителями напряжения действенна при кратковременных перенапряжениях. При длительных – токи короткого замыкания могут вывести ограничители из строя, и устройства на линии окажутся без защиты. Для защиты от коротких замыканий в линию можно последовательно включить плавкие предохранители.

5.10. Дополнительные меры защиты от помех

Диагностика. Если есть возможность выбора маршрута прокладки кабеля с замером уровня помех – не стоит ей пренебрегать. Даже если программная коррекция ошибок успешно справляется со сбоями, нужно сделать все, чтобы физически снизить уровень помех в линии. Полезно предусмотреть в программе диагностический режим, в котором накапливалась бы статистика сбоев, обрабатываемых программной коррекцией (провал по контрольной сумме или тайм-ауту). Если сбоев слишком много, желательно поработать над поиском и устранением их причины. Снижение скорости связи (бодрейта) во многих случаях повышает помехоустойчивость. Не имеет смысла устанавливать скорость обмена больше, чем необходимо для нормальной работы системы, если только не требуется запас на модификацию.

Прокладка кабеля. По возможности не следует проводить витую пару вдоль силовых кабелей, тем более в общей оплетке, так как существует опасность наводок от силовых токов через взаимную индуктивность. Силовое оборудование, коммутирующее большие токи, также является источником помех. Сигнальные цепи питания оптоизолированной линии

лучше не использовать для питания чего-либо еще, так как протекающие по сигнальной "земле" лишние токи могут вносить в линию дополнительный шум. Некачественная витая пара с асимметричными характеристиками проводников – еще один источник проблем. Чем меньше шаг витой пары (чаще перевиты провода) тем лучше. Даже если не применяется опто-развязанная линия или дренаж, стоит сразу провести кабель с запасной витой парой – на случай, если произойдет обрыв первой или все же понадобится провести сигнальную «землю».

Экранирование и заземление. В промышленных условиях, тяжелых в плане электромагнитного шума, рекомендуется применять экранированный кабель с витой парой. Экран, охватывающий проводники линии, защищает их от паразитных емкостных связей и внешних магнитных полей. Экран следует заземлять только в одной из крайних точек линии. Заземление в нескольких точках недопустимо: из-за разности потенциалов местных "земель" по экрану могут протекать существенные токи, которые будут создавать наводки на сигнальные проводники. Некоторые разработчики рекомендуют для защиты от радиопомех дополнительно включать в нескольких местах между экраном и заземлением специальные высокочастотные конденсаторы емкостью 1...10 нФ.

Индуктивные фильтры (рис. 5.16). Если в линию все же попадают высокочастотные помехи, их можно отсеять индуктивными фильтрами. Существуют специальные индуктивные фильтры, предназначенные для подавления высокочастотных помех в линиях связи. Они последовательно включаются в линию непосредственно у приемников. Например, микросхема В82790-S фирмы Еrcos выполнена в виде четырехполюсника, через который витая пара подсоединяется к приемнику.

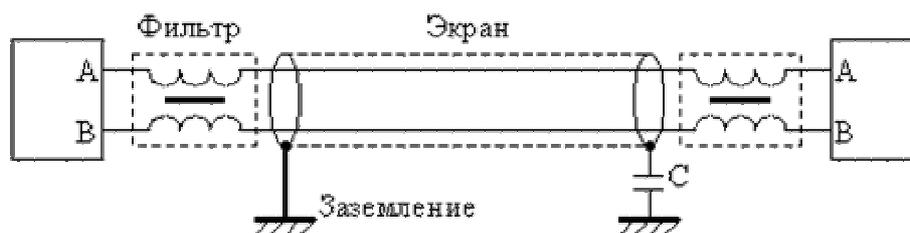


Рис. 5.16. Экранирование линии связи

Интерфейс RS-485 наиболее часто используется при создании современных локальных сетей различного назначения как в промышленных изделиях, так и в любительской практике. Основными преимуществами интерфейса являются:

- Относительно низкая себестоимость микросхем драйверов, что снижает стоимость аппаратной реализации сетевых диспетчеров, то есть узлов связи между сетевой средой (линиями связи) и ядром станции (узлом сети).

- Использование в сетях на базе интерфейса RS-485 всего трех проводов (третий, общий, не всегда является обязательным), что значительно снижает себестоимость всей системы, поскольку известно, что себестоимость сетевой среды современных локальных сетей практически всегда составляет более 60 % от стоимости всей системы.

- Микросхемы драйверов имеют малые габаритные размеры. Наиболее часто используются микросхемы, выполненные в корпусе DIP8 со стандартным расположением выводов, ставшим de facto в промышленном стандарте. Микросхемы драйверов используют всего несколько дискретных элементов для цепей защиты, использование которых не является обязательным. Малые габаритные размеры микросхем драйверов и минимальное количество обвязки экономит площадь печатной платы, что также положительно сказывается на стоимости системы.

- Современные микросхемы имеют достаточно низкое энергопотребление, многие из них при отсутствии активности в сети автоматически переходят в режим экономии, что снижает энергопотребление системы.

- Современные микросхемы драйверов имеют повышенную нагрузочную способность. Если ранее большинство микросхем было рассчитано на работу с 32 станциями, то современные модели обеспечивают нормальное функционирование до 256 станций.

- В настоящее время выпускаются микросхемы с высокой предельной скоростью передачи. Это позволяет создавать высокоскоростные сети и снижает количество ошибок в сети за счет улучшения формы передаваемого сигнала.

- Драйверы интерфейса RS-485 имеют достаточно простое управление. Все микросхемы драйверов можно условно разделить на 4 группы: микросхемы с питанием +5 В, микросхемы с расширенным диапазоном питания от 3 до 5,5 В, низковольтные микросхемы с питанием 3,3 В и микросхемы со встроенной оптической изоляцией.

Определим критерии отбора микросхем для последующего рассмотрения. Мы ставим своей целью ознакомление читателя с широко используемыми микросхемами интерфейса RS-485 (но не RS-422), то есть с микросхемами, работающими в полудуп-

лексном режиме. У этих микросхем входы приемника объединены с выходами передатчика и образуют две линии приема/передачи *A* и *B*. Мы не будем рассматривать ряд микросхем, содержащих только приемники или только передатчики, поскольку их применение также весьма ограничено. И наконец, мы будем рассматривать только микросхемы, выпускаемые в корпусе с восемью выводами, как наиболее распространенные.

Первая группа микросхем самая многочисленная, она насчитывает 60 типов. Микросхемы этой группы предназначены для систем со стандартным питанием. С учетом критериев, приведенных выше, остается для рассмотрения только 20 типов микросхем. Одними из первых начали производиться микросхемы MAX481/483/485/487. Затем к этому семейству добавилась микросхема MAX1487, а намного позже эти микросхемы были оснащены цепями защиты от электростатики до ± 15 кВ и к обозначению этих микросхем была добавлена буква «Е». Аналогично и у других микросхем наличие буквы «Е» после обозначения означает встроенную защиту от электростатики. В общем, эти микросхемы имели не очень хорошие показатели, конечно, по сравнению с современными микросхемами. Они позволяли объединять в сеть только 32 интерфейса (за исключением MAX487, который мог объединять до 128 станций) и обеспечивали не очень высокую скорость передачи данных. Однако именно

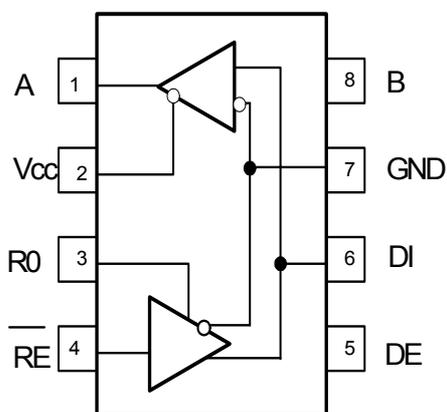


Рис. 5.17. Расположение выводов семейства MAX485

они и их аналоги легли в основу стандартного расположения выводов микросхем интерфейса.

Напомним обозначения выводов (рис. 5.17):

R0 – receiver output – выход приемника. Если $A > B$ на 200 мВ, то $R0 = 1$, если $A < B$ на 200 мВ – $R0 = 0$.

RE – receiver output enable – разрешение выхода приемника при $RE = 0$. При $RE = 1$ выход R0 находится в высокоимпедансном состоянии.

DE – driver output enable – разрешение выходов передатчика. Если $DE = 1$, выходы активны, в противном случае они находятся в высокоимпедансном состоянии.

DI – driver input – вход передатчика.

GND – ground – общий провод питания.

A – no inverting receiver input and driver output – неинвертирующий вход/выход.

B – inverting receiver input and driver output – инвертирующий вход/выход.

V_{CC} – positive supply – напряжение питания.

Семейство микросхем MAX3082/3085/3088 по выводам полностью совместимо со стандартным семейством MAX481/483/485/487/1487. Основными отличиями этой группы являются:

- Повышенная нагрузочная способность выходов, что позволяет объединять в сеть до 256 станций.
- Наличие защиты от электростатики для микросхем с буквой «E».
- Наличие режима экономии, в который микросхемы переходят при закрытии приемника, то есть при RE = 1.
- Повышенное быстродействие: 500 кбит/с, для MAX3085(E) и 10 Мбит/с для MAX3088(E).

Семейство микросхем MAX3463/3464 по выводам также полностью совместимо со стандартным семейством MAX481/483/485/487/1487. Основными отличиями этой группы являются:

- Высокое быстродействие – до 20 Мбит/с.
- В микросхеме MAX3463 имеется схема автоматического определения неправильного подключения линий A и B.
- Средняя нагрузочная способность выходов, что позволяет объединять в сеть до 128 станций.
- Отсутствие защиты от электростатики.
- Наличие режима экономии, в который микросхемы переходят при выключении приемника и передатчика через 50 нс, а также в случае включенного приемника при статическом состоянии входов более 800 нс. Ток потребления в режиме экономии не превышает 1 мкА.
- Для микросхем этого семейства гарантируется высокий уровень на выходе приемника, если линии сети A и B замкнуты или свободны или все подключенные к линиям передатчики выключены.

- Микросхемы оснащены цепями защиты входов от «горячего» включения в шину сети.

- Они имеют защиту выходов передатчиков от короткого замыкания или температуры выходных каскадов передатчика более +140 °С.

- Имеют типовое время переключения приемника не более 2нс.

- Микросхемы совместимы с промышленным стандартом PROFIBUS, в котором используются два терминатора по 220 Ом на концах сети и подтягивающие резисторы по 390 Ом. При этом гарантируется размах уровней линий А и В не менее 2,1 В.

Еще одна интересная микросхема этой группы MAX1483 разработана специально для систем со сверхмалым энергопотреблением. Разводка ее выводов также совпадает со стандартной. Микросхема обладает средним быстродействием 250 кбит/с, током потребления в рабочем режиме не более 20 мкА и обеспечивает связь с 256 станциями.

Представляет также интерес микросхема MAX3443E, предназначенная для работы в сетях с повышенным уровнем помех. Перечислим основные достоинства этой микросхемы:

- Высокое быстродействие – до 10 Мбит/с.

- Наличие режима экономии, в который микросхемы переходят при выключении приемника и передатчика через 50 нс, а также в случае включенного приемника при статическом состоянии входов более – 800 нс. Ток потребления в режиме экономии не превышает 10 мкА.

- Наличие защиты от электростатики до ±15 кВ.

- Микросхема имеет встроенную защиту от импульсных помех в линии до +60 В. Защита гарантируется независимо от состояния микросхемы и наличия или отсутствия ее питания.

- Микросхема совместима со стандартом J1708.

Другие достоинства микросхемы MAX3443E аналогичны достоинствам MAX3463, перечисленным выше.

Напомним читателю, что стандартная конфигурация сети RS-485 выглядит, как показано на рис. 5.18, *а*. Управление драйверами осуществляется входами RE и DE, а передача информации осуществляется через DI и RO. Стандарт J1708 предполагает, что приемник всегда открыт, то есть RE соединен с общим проводом GND (рис. 5.18, *б*). Вход данных DI передат-

чика также соединен с общим проводом GND, а передача данных осуществляется по входу управления DE через инвертор. Таким образом, весь обмен осуществляется только двумя сигналами. Еще одной особенностью стандарта является использование изолированной «земли» в шине, как показано на рис. 5.18, б.

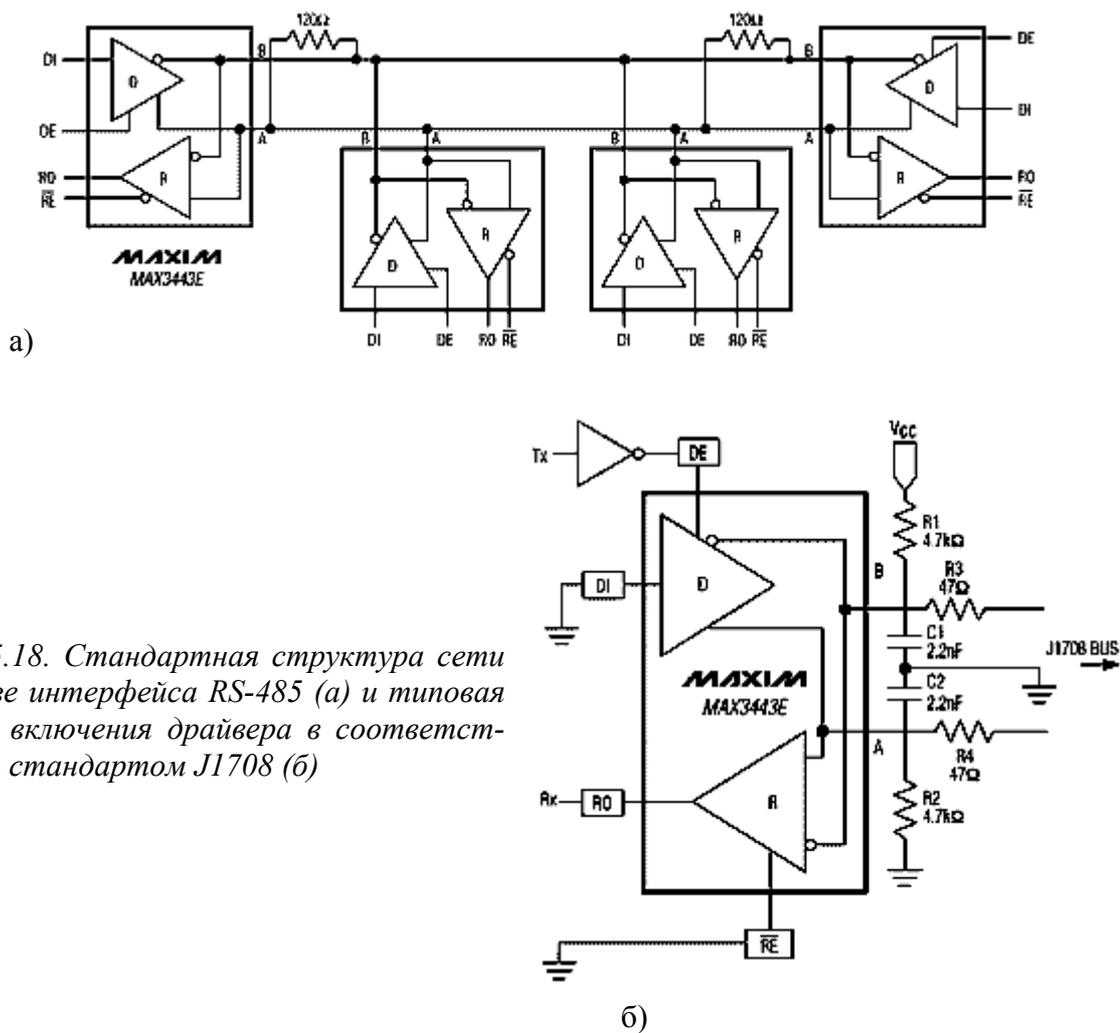


Рис. 5.18. Стандартная структура сети на базе интерфейса RS-485 (а) и типовая схема включения драйвера в соответствии со стандартом J1708 (б)

Приемник в этой группе микросхем всегда открыт. Микросхема MAX3280 содержит только приемник и представляет ограниченный интерес. В остальных микросхемах управление передатчиком осуществляется разными уровнями для различных микросхем. Микросхемы имеют встроенную защиту от электростатики и режим энергосбережения.

На рис. 5.19 приведена типовая схема включения микросхем семейства MAX1480. Для семейства MAX3480 схема включения аналогична и отличается только номиналами некоторых резисторов.

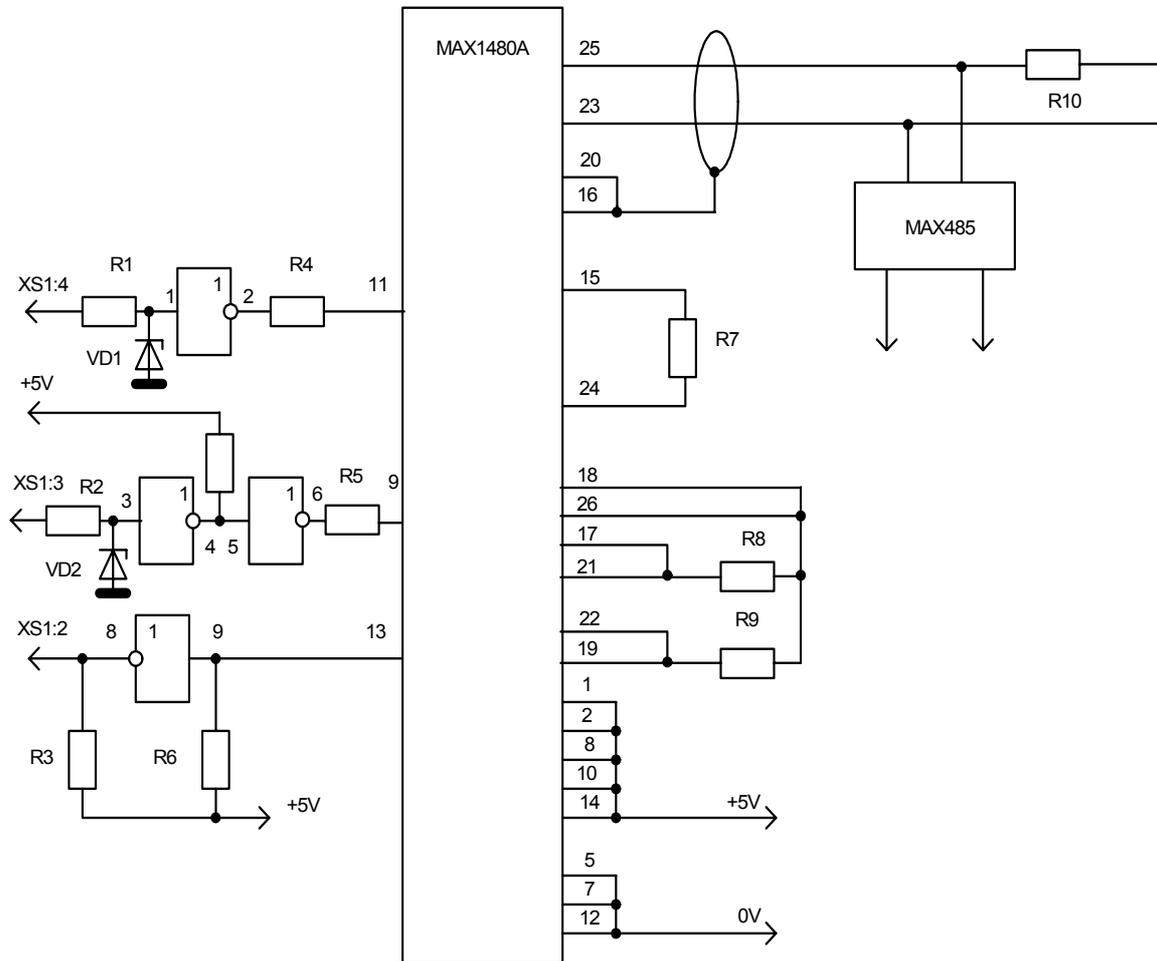


Рис. 5.19. Схема включения драйвера MAX1480A

В заключение отметим, что с точки зрения идентичности схемотехники для систем с различным питанием и с точки зрения экономии площади печатной платы наиболее предпочтительно использовать микросхемы серии MAX3280(E)/3281(E)/3283(E)/3284(E) с расширенным диапазоном питания. Однако существует ряд причин, способных привести к выбору другой элементной базы. Например, необходимо использовать наиболее дешевые микросхемы или микросхемы с более высокой нагрузочной способностью. В каждом конкретном случае в процессе разработки необходимо учитывать все требования и исходя из них выбирать требуемую элементную базу.

Контрольные вопросы

1. Назовите наиболее употребительные коммуникационные каналы последовательной передачи данных.
2. Как представляется информация в кадре, передаваемом по последовательному каналу?
3. Есть ли отличия в физической реализации каналов по стандарту RS-232 и RS-485 и в чем они заключаются?
4. Охарактеризуйте разновидности последовательных каналов по способу передачи данных и физической реализации.
5. В чем особенность асинхронного и синхронного способов передачи данных?
6. Назовите все функции, реализуемые стандартом RS-232.
7. Приведите разновидности микросхем, реализующих протокол RS-232, и назовите их основные характеристики.
8. Какие функции реализует режим прерываний в стандартных микросхемах?
9. Функции буфера FIFO.
10. Перечислите назначения бит регистра состояния линии.
11. В чем принципиальное отличие протокола RS-485 от RS-232C?
12. Опишите функции приемопередатчика для RS-485 и расскажите о способе сопряжения его с микропроцессором или персональной ЭВМ.
13. В чем заключается особенность работы линий связи и для чего выполняется режим согласования линии?
14. Какое схемотехническое решение можно применять для организации согласования линий передачи информации при обеспечении заданной надежности?
15. Дайте понятие безотказности и истинной безотказности и поясните особенность построения линий связи на примере.
16. Разрешается ли «горячее» подключение к линиям передачи данных?
17. Какие мероприятия надо выполнять для снижения влияния помех в линиях связи?
18. Каким образом можно различать передаваемые команды и данные?
19. Можно ли использовать девятый бит для распознавания передаваемых данных и команд и что нужно для этого сделать?
20. Как компенсировать влияние помехи при приеме информации?
21. В каких случаях используется контрольная сумма и что она означает?
22. Что означает режим «квитирование» и в каких случаях его необходимо использовать?
23. Как защитить приемопередатчик от разрушения?
24. В каких случаях используется экранирование, а в каких заземление?
25. Приведите несколько примеров типовых схем преобразования сигналов различных фирм производителей.

Глава 6. ПРОТОКОЛЫ CAN И PROFIBUS

6.1. CAN-протокол

CAN-протокол был разработан фирмой Robert Bosch GmbH для использования в автомобильной электронике, отличается повышенной помехоустойчивостью, надежностью и обладает следующими возможностями:

- конфигурационная гибкость,
- получение сообщений всеми узлами с синхронизацией по времени,
- неразрушающий арбитраж доступа к шине,
- режим «мультимастер»,
- обнаружение ошибок и передача сигналов об ошибках,
- автоматическая передача сбойных сообщений при получении возможности повторного доступа к шине,
- различие между случайными ошибками и постоянными отказами узлов с возможностью выключения дефектных узлов,
- работает по витой паре на расстоянии до 1 км.

Физический уровень протокола CAN

Физический уровень (physical layer) протокола CAN определяет сопротивление кабеля, уровень электрических сигналов в сети и т.п. Существует несколько физических уровней протокола CAN (ISO 11898, ISO 11519, SAE J2411).

В подавляющем большинстве случаев используется физический уровень CAN, определенный в стандарте ISO 11898. В качестве среды передачи ISO 11898 определяет двухпроводную дифференциальную линию с импедансом (терминаторы) 120 Ом (допускается колебание импеданса в пределах от 108 до 132 Ом). Физический уровень CAN реализован в специальных чипах – CAN-приемопередатчиках (transceivers), которые преобразуют обычные TTL-уровни сигналов, используемых CAN-контроллерами в уровни сигналов на шине CAN (рис. 6.1). Наиболее распространенный CAN-приемопередатчик – Phillips 82C250, который полностью соответствует стандарту ISO 11898.

Максимальная скорость сети CAN в соответствии с протоколом равна 1 Мбит/с. При данной скорости максимальная длина кабеля равна при-

мерно 40 м. Ограничение на длину кабеля связано с конечной скоростью света и механизмом побитового арбитража (во время арбитража все узлы сети должны получать текущий бит передачи одновременно, то есть сигнал должен успеть распространиться по всему кабелю за единичный отсчет времени в сети). Соотношение между скоростью передачи и максимальной длиной кабеля приведено в табл. 6.1.

Таблица 6.1.

Соотношение между скоростью передачи и максимальной длиной кабеля

Скорость, кбит/с	1000	500	250	125	10
Максимальная длина сети, м	40	100	200	500	600

Разъемы для сети CAN до сих пор не стандартизованы. Каждый протокол высокого уровня обычно определяет свой тип разъемов для CAN-сети.

Естественно, что все эти качества делают CAN-протокол весьма привлекательным для использования в производственных приложениях, тем более что он поддерживается рядом фирм-производителей микросхем, выпускающих недорогие устройства, которые аппаратно реализуют требования CAN-протокола и работают в широком температурном диапазоне.

CAN-протокол распространяется на следующие уровни:

- *Объектный уровень* обеспечивает фильтрацию сообщений и обработку сообщений и состояний.

- *Транспортный уровень* представляет собой ядро CAN-протокола. Он отвечает за синхронизацию, арбитраж, доступ к шине, разделение посылок на фреймы, определение и передачу ошибок и минимизацию неисправностей.

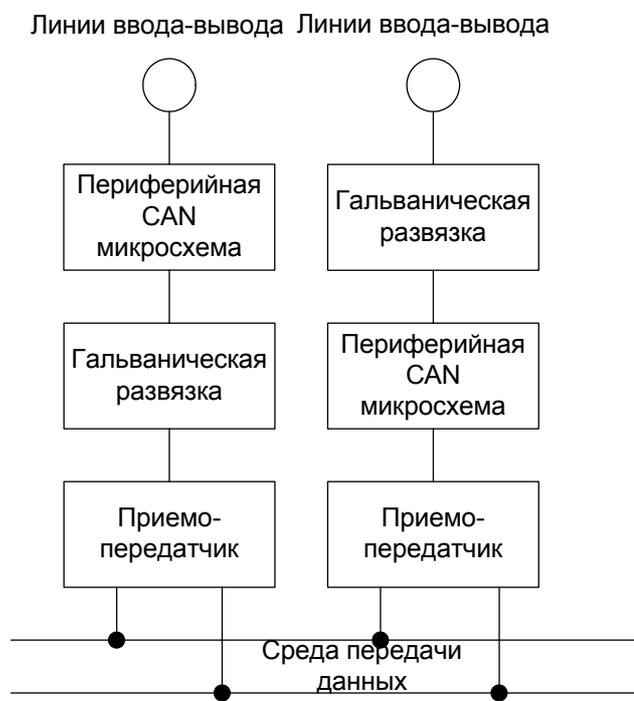


Рис. 6.1. Способы реализации линий ввода-вывода

- *Физический уровень* определяет, как именно будут передаваться сигналы, их электрические уровни и скорость передачи ввода/вывода.

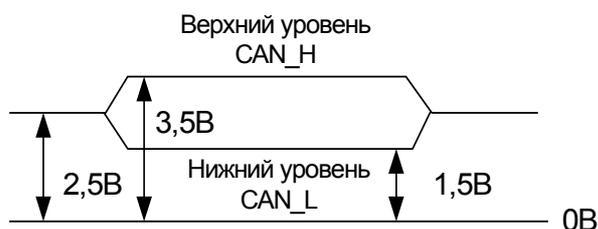


Рис. 6.2. Типичные значения напряжения питания

приемопередающим буфером и микросхемой, обеспечивающей функции CAN, либо между микросхемой и остальной системой (рис. 6.3).

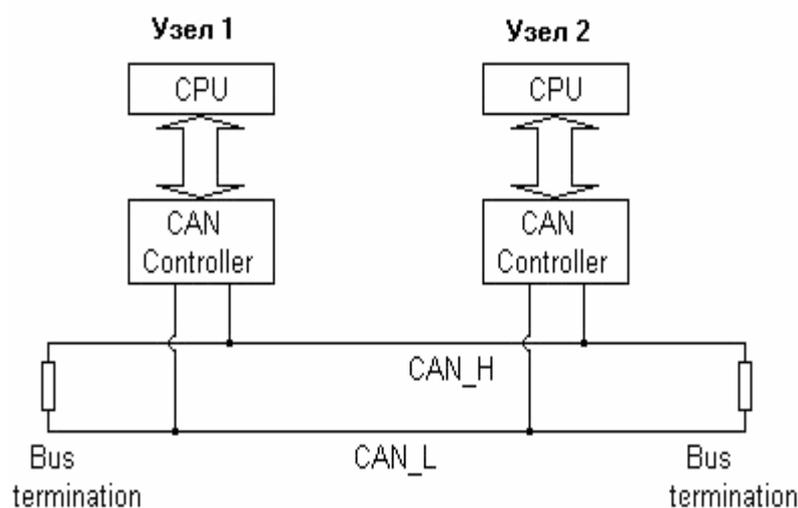


Рис. 6.3. Промышленная сеть реального времени CAN

весь трафик, передаваемый по шине. Однако CAN-контроллеры предоставляют аппаратную возможность фильтрации CAN-сообщений.

Каждый узел состоит из двух составляющих. Это собственно CAN-контроллер, который обеспечивает взаимодействие с сетью и реализует протокол, и микропроцессор (CPU).

CAN-контроллеры соединяются с помощью дифференциальной шины, которая имеет две линии – CAN_H (can-high) и CAN_L (can-low), по которым передаются сигналы. Логический ноль регистрируется, когда на линии CAN_H сигнал выше, чем на линии CAN_L, логическая единица – когда сигналы CAN_H и CAN_L одинаковы (отличаются менее чем на 0,5 В). Использование такой дифференциальной схемы передачи делает возмож-

Например, типичные значения при напряжении питания +5 В приведены на рис. 6.2, причем доминирующим является нижний уровень, а рецессивным, соответственно, верхний.

Верхний уровень (CAN_H) 3,5 В оценивается либо между

Промышленная сеть реального времени CAN представляет собой сеть с общей средой передачи данных. Это означает, что все узлы сети одновременно принимают сигналы, передаваемые по шине. Невозможно послать сообщение какому-либо конкретному узлу. Все узлы сети принимают

ным работу CAN-сети в очень сложных внешних условиях. Логический ноль называется доминантным битом, а логическая единица – рецессивным. Эти названия отражают приоритет логической единицы и нуля на шине CAN. При одновременной передаче в шину логического нуля и единицы, на шине будет зарегистрирован только логический ноль (доминантный сигнал), а логическая единица будет подавлена (рецессивный сигнал).

CAN open – это открытая промышленная сеть, созданная на основе Controller Area Network (CAN). Стандарт CAN (ISO11898) описывает два нижних уровня эталонной модели ISO/OSI, CAN open описывает остальные пять. Документ The CAN open Application Layer and Communication Profile (CiA DS 301) определяет, каким образом устройства обмениваются данными, и описывает интерфейс к нижележащим уровням сети.

Общая схема связи устройств в CAN open. Протокол CAN open определяет несколько методов передачи сообщений по сети CAN. Эти сообщения называются объектами связи (communication objects). CAN open поддерживает синхронизованную передачу сообщений, которая обеспечивается объектами Sync и Time Stamp. Асинхронные сообщения (или события) могут пересылаться в любой момент времени. В целом CAN open определяет четыре типа сообщений (communication objects):

- сообщения управления сетью, например Layer Management (LMT) и Network Management (NMT) сообщения;
- так называемые Service Data Objects (SDO);
- так называемые Process Data Objects (PDO);
- предопределенные сообщения (Sync Object, Time Stamp Object, Emergency Object).

Инициализация и управление сетью. Сервис управления сетью используется для контроля состояния устройств в сети CAN open. В рамках сервиса управления сетью доступны следующие функции:

- динамическое или статическое распределение идентификаторов CAN для соединений SDO/PDO и сервиса обработки ошибок;
- управление состоянием работы устройств и контроль режимов соединений в устройствах;
- периодический опрос для определения сбоев в устройствах;
- вместо опроса каждое устройство может периодически посылать сообщение о том, что оно функционирует нормально.

Механизм передачи данных. CAN open определяет два совершенно разных механизма передачи данных.

Service Data Object (SDO) обычно используется для конфигурирования устройств низкой приоритетности. Отдельные параметры устройства адресуются при помощи 16-битного адреса и 8-битного подадреса. С помощью SDO можно передавать данные длиной больше восьми байт, используя механизм фрагментации. Функциональность SDO:

- передача данных любого размера,
- чтение и запись любых данных с подтверждением,
- быстрая передача данных длиной до 4 байт,
- обрыв соединения с любого конца с передачей ошибки через сеть.

Все параметры устройства объединены в object dictionary (словарь объектов), и все объекты в object dictionary могут быть прочитаны или изменены удаленно при помощи SDO.

Process Data Object (PDO) используется для передачи с высокой скоростью высокоприоритетных данных, так как PDO-сообщения не содержат никаких дополнительных протокольных данных. При помощи PDO можно передавать только данные, длина которых меньше 8 байт. Формат данных PDO может быть фиксированным или может быть сконфигурирован при помощи SDO. PDO-сообщения могут быть переданы одним узлом сразу нескольким другим узлам одновременно.

События. CAN open поддерживает несколько способов передачи данных реального времени.

При возникновении какого-либо события можно послать PDO-сообщение. Например, устройство дискретного ввода-вывода может отсылать состояние своих выводов в сеть при их изменении. Такой способ позволяет минимизировать загрузку сети и увеличить ее пропускную способность.

Возможен синхронный режим передачи данных. В этом режиме устройства синхронизируют передачу данных в сеть с часами Master-устройства. Этот режим особенно полезен, когда контуры управления замыкаются через сеть (так называемые сетевые системы управления).

Кроме перечисленных выше способов передачи данных, можно использовать передачу по запросу (polling). В любой момент можно использовать PDO-сообщение для инициации передачи данных устройством. Эта схема использует RTR-бит CAN-кадра.

Логический уровень протокола CAN

Данные в CAN передаются короткими сообщениями-кадрами стандартного формата. В CAN существуют четыре типа сообщений: Data Frame, Remote Frame, Error Frame, Overload Frame.

Data Frame – это наиболее часто используемый тип сообщения. Он состоит из следующих основных частей (рис. 6.4):

- поле арбитража (arbitration field) определяет приоритет сообщения в случае, когда два или более узлов одновременно пытаются передать данные в сеть. Поле арбитража состоит, в свою очередь: из 11-битного идентификатора + 1 бит RTR (retransmit) для стандарта CAN-2,0A, из 29-битного идентификатора + 1 бит RTR (retransmit) для стандарта CAN-2,0B. Следует отметить, что поле идентификатора, несмотря на свое название, никак не идентифицирует само по себе ни узел в сети, ни содержимое поля данных. Для Data-кадра бит RTR всегда выставлен в логический ноль (доминантный сигнал);

- поле данных (data field) содержит от 0 до 8 байт данных;
- поле CRC (CRC field) содержит 15-битную контрольную сумму сообщения, которая используется для обнаружения ошибок;

- слот подтверждения (Acknowledgement Slot) (1 бит). Каждый CAN-контроллер, который правильно принял сообщение, посылает бит подтверждения в сеть. Узел, который послал сообщение, слушает этот бит, и в случае если подтверждение не пришло, повторяет передачу. В случае приема слота подтверждения передающий узел может быть уверен лишь в том, что хотя бы один из узлов в сети правильно принял его сообщение.

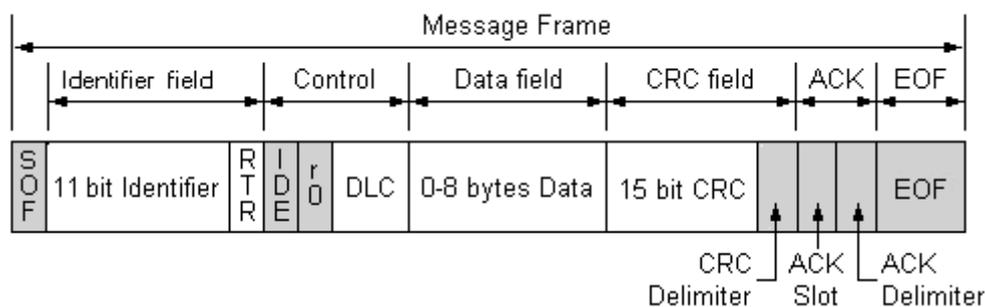


Рис. 6.4. Data frame стандарта CAN 2.0A

Remote Frame – это Data Frame без поля данных и с выставленным битом RTR (1 - рецессивные бит). Основное предназначение Remote-кадра –

это инициация одним из узлов сети передачи в сеть данных другим узлом. Такая схема позволяет уменьшить суммарный трафик сети. Однако на практике Remote Frame сейчас используется редко (например, в DeviceNet Remote Frame вовсе не используется).

Error Frame – это сообщение, которое явно нарушает формат сообщения CAN. Передача такого сообщения приводит к тому, что все узлы сети регистрируют ошибку формата CAN-кадра и в свою очередь автоматически передают в сеть Error Frame. Результатом этого процесса является автоматическая повторная передача данных в сеть передающим узлом. Error Frame состоит из поля Error Flag, которое состоит из 6 бит одинакового значения (и таким образом Error frame нарушает проверку Bit Stuffing, см. ниже), и поля Error Delimiter, состоящего из 8 рецессивных битов. Error Delimiter дает возможность другим узлам сети, обнаружив Error Frame, послать в сеть свой Error Flag.

Overload Frame – повторяет структуру и логику работы Error-кадра, с той разницей, что он используется перегруженным узлом, который в данный момент не может обработать поступающее сообщение и поэтому просит при помощи Overload-кадра о повторной передаче данных. В настоящее время Overload-кадр практически не используется.

Контроль доступа к среде передачи (побитовый арбитраж). Поле арбитража CAN-кадра используется в CAN для разрешения коллизий доступа к шине методом недеструктивного арбитража. Суть метода недеструктивного арбитража заключается в следующем. В случае, когда несколько контроллеров начинают одновременную передачу CAN-кадра в сеть, каждый из них сравнивает бит, который собирается передать на шину, с битом, который пытается передать на шину, конкурирующий контроллер (рис. 6.5). Если значения этих битов равны, оба контроллера передают следующий бит. И так происходит до тех пор, пока значения передаваемых битов не окажутся различными. Теперь контроллер, который передавал логический ноль (более приоритетный сигнал), будет продолжать передачу, а другой (другие) контроллер прервет свою передачу до того времени, пока

шина вновь не освободится. Конечно, если шина в данный момент занята, то контроллер не начнет передачу до момента её освобождения.

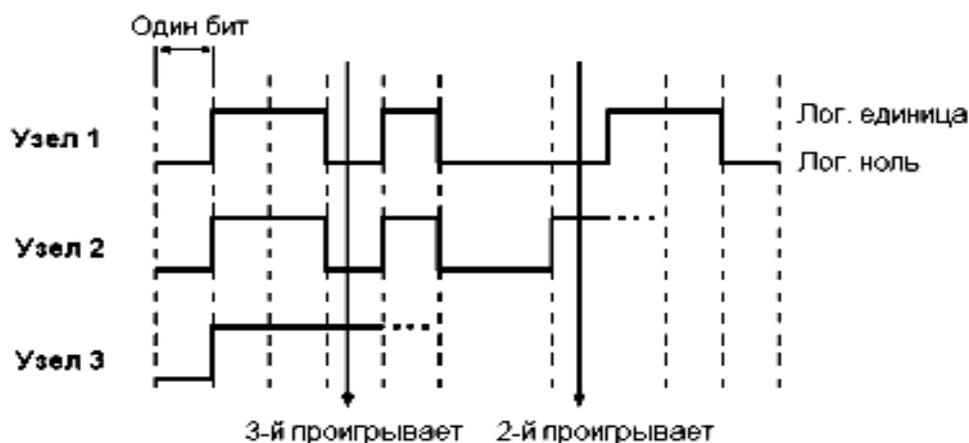


Рис. 6.5. Побитовый арбитраж на шине CAN

Методы обнаружения ошибок. CAN-протокол определяет пять способов обнаружения ошибок в сети:

- Bit monitoring,
- Bit stuffing ,
- Frame Check,
- ACKnowledgement Check,
- CRC Check.

Bit monitoring. Каждый узел во время передачи битов в сеть сравнивает значение передаваемого им бита со значением бита, которое появляется на шине. Если эти значения не совпадают, то узел генерирует ошибку Bit Error. Естественно, что во время арбитража на шине (передача поля арбитража в шину) этот механизм проверки ошибок отключается.

Bit stuffing. Когда узел передает последовательно в шину 5 бит с одинаковым значением, то он добавляет шестой бит с противоположным значением. Принимающие узлы этот дополнительный бит удаляют. Если узел обнаруживает на шине больше 5 последовательных бит с одинаковым значением, то он генерирует ошибку Stuff Error.

Frame Check. Некоторые части CAN-сообщения имеют одинаковое значение во всех типах сообщений. То есть протокол CAN точно определяет, какие уровни напряжения и когда должны появляться на шине. Если формат сообщений нарушается, то узлы генерируют ошибку Form Error.

ACKnowledgement Check. Каждый узел, получив правильное сообщение по сети посылает в сеть доминантный (0) бит. Если же этого не происходит, то передающий узел регистрирует ошибку Acknowledgement Error.

CRC Check. Каждое сообщение CAN содержит CRC-сумму, и каждый принимающий узел подсчитывает значение CRC для каждого полученного сообщения. Если подсчитанное значение CRC-суммы не совпадает со значением CRC в теле сообщения, принимающий узел генерирует ошибку CRC Error.

Механизм ограничения ошибок (Error confinement). Каждый узел сети CAN во время работы пытается обнаружить одну из пяти возможных ошибок. Если ошибка обнаружена, узел передает в сеть Error Frame, разрушая тем самым весь текущий трафик сети (передачу и прием текущего сообщения). Все остальные узлы обнаруживают Error Frame и принимают соответствующие действия (сбрасывают принятое сообщение). Кроме того, каждый узел ведет два счетчика ошибок: Transmit Error Counter (счетчик ошибок передачи) и Receive Error Counter (счетчик ошибок приема). Эти счетчики увеличиваются или уменьшаются в соответствии с несколькими правилами. Сами правила управления счетчиками ошибок достаточно сложны, но сводятся к простому принципу: ошибка передачи приводит к увеличению Transmit Error счетчика на 8, ошибка приема увеличивает счетчик Receive Error на 1, любая корректная передача (прием) сообщения уменьшает соответствующий счетчик на 1. Эти правила приводят к тому, что счетчик ошибок передачи передающего узла увеличивается быстрее, чем счетчик ошибок приема принимающих узлов. Это правило соответствует предположению о большой вероятности того, что источником ошибок является передающий узел.

Каждый узел CAN-сети может находиться в одном из трех состояний. Когда узел стартует, он находится в состоянии Error Active. Когда значение хотя бы одного из двух счетчиков ошибок превышает предел 127, узел переходит в состояние Error Passive. Когда значение хотя бы одного из двух счетчиков превышает предел 255, узел переходит в состояние Bus Off.

Узел, находящийся в состоянии Error Active, в случае обнаружения ошибки на шине передает в сеть Active Error Flags. Active Error Flags состоит из 6 доминантных бит, поэтому все узлы его регистрируют. Узел в состоянии Passive Error передает в сеть Passive Error Flags при обнаружении ошибки в сети. Passive Error Flags состоит из 6 рецессивных бит, по-

этому остальные узлы сети его не замечают, и Passive Error Flags лишь приводит к увеличению Error-счетчика узла. Узел в состоянии Bus Off ничего не передает в сеть (не только Error-кадры, но вообще никакие другие).

Адресация и протоколы высокого уровня. В CAN не существует явной адресации сообщений и узлов. Протокол CAN нигде не указывает, что поле арбитража (Identification field + RTR) должно использоваться как идентификатор сообщения или узла. Таким образом, идентификаторы сообщений и адреса узлов могут находиться в любом поле сообщения (в поле арбитража или в поле данных или присутствовать и там и там). Точно так же протокол не запрещает использовать поле арбитража для передачи данных.

Утилизация поля арбитража и поля данных и распределение адресов узлов, идентификаторов сообщений и приоритетов в сети является предметом рассмотрений так называемых протоколов высокого уровня (HLP – Higher Layer Protocols). Название HLP отражает тот факт, что протокол CAN описывает только два нижних уровня эталонной сетевой модели ISO/OSI, а остальные уровни описаны протоколами HLP.

Существует множество таких высокоуровневых протоколов. Наиболее распространенные из них: Device Net, CAL/CAN open, SDS, Can Kingdom .

Семейство протоколов CAN

Промышленная сеть CAN (Controller Area Network) была создана в конце 80-х годов фирмой Bosch как решение для распределенных систем, работающих в режиме реального времени. Первая реализация CAN использовалась в автомобильной электронике, однако сейчас CAN находит применение практически в любых типах машин и промышленных установок, от простейших бытовых приборов до систем управления ускорителями элементарных частиц. В настоящий момент CAN-протокол стандартизован в международном стандарте ISO 11898.

Основные положения стандарта CAN:

- В качестве среды передачи в CAN используется дифференциальная линия связи – витая пара, сигналы по которой передаются в дифференциальном режиме.
- Для контроля доступа к среде передачи используется метод неструктивного арбитража.
- Данные передаются короткими (максимальная длина поля данных 8 байт) пакетами, которые защищены контрольной суммой.

- В CAN отсутствует явная адресация сообщений. Вместо этого каждый пакет снабжен полем арбитража (идентификатор + RTR-бит), которое задает приоритет сообщения в сети.

- В случае возникновения ошибок передачи/приема сообщения CAN имеет исчерпывающую схему контроля ошибок, которая гарантирует повторную передачу пакета.

- В CAN существует способ автоматического устранения узла, являющегося источником ошибочных пакетов в сети.

Протоколы высокого уровня (HLP). Протокол CAN описывает только то, как пакеты должны быть доставлены от одного узла сети к другому. CAN ничего не говорит о том, как нужно интерпретировать поле данных пакета, утилизировать поле арбитража, обеспечить передачу данных, длина которых превышает 8 байт, какую логическую схему передачи должны использовать общающиеся между собой узлы и т.п. Другими словами, CAN описывает только два нижних уровня эталонной сетевой модели ISO/OSI.

Положения, которые не специфицируются стандартом CAN (верхние пять уровней модели ISO/OSI), описываются так называемыми CAN-протоколами высокого уровня (HLP – Higher Layer Protocols). В настоящий момент существует несколько HLP-протоколов. Они описывают следующие общие положения:

- процедуру подключения узла к сети;
- способ утилизации поля арбитража (распределение уровней приоритетов и идентификаторов между узлами сети);
- формат сообщений;
- схему обработки ошибок на логическом уровне;
- логические схемы обмена данными между узлами сети (обмен данными в режиме master-slave, обмен данными с установлением соединения и без него и т.п.).

Протокол CAN полностью реализован в кремниевых чипах – CAN-контроллерах. Сегодня на рынке существуют сотни CAN-контроллеров производимых различными фирмами.

В силу своих особенностей (побитовый арбитраж, дифференциальный режим передачи сигналов, исчерпывающая схема обработки ошибок) промышленная сеть CAN хорошо подходит для создания распределенных систем сбора данных, систем контроля и управления.

Типы фреймов в CAN-протоколе

В CAN-протоколе определены следующие типы фреймов:

- фрейм данных перемещает данные с передатчика на приемник (приемники);
- удаленный фрейм запрашивает передачу фрейма данных, связанного с определенным идентификатором;
- фрейм ошибки выражает, какой узел обнаружил ошибку шины/сети;
- фрейм перегрузки обеспечивает задержку между передачей фреймов, чтобы управлять потоком данных.

Рассмотрим подробнее фрейм данных (рис. 6.6).

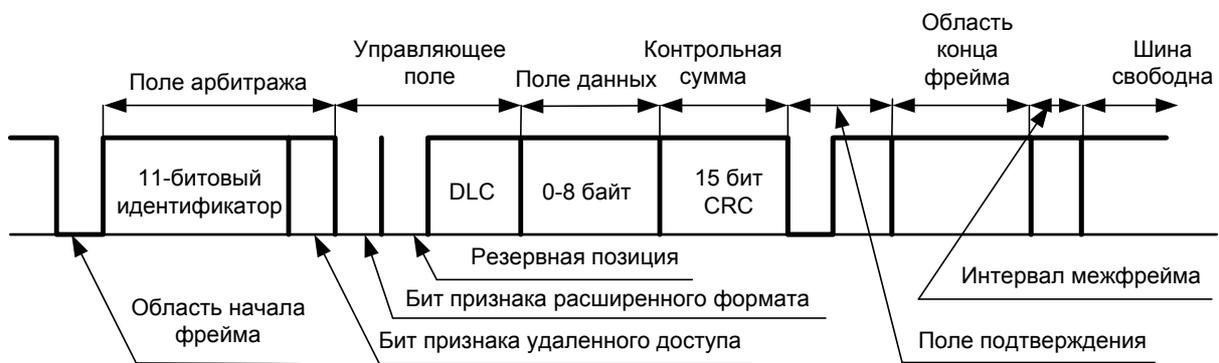


Рис. 6.6. Управляющее поле фрейма данных

Стандартный формат (Standard Format) состоит из стартового поля SOF, поля арбитража Arbitration Field, управляющего поля Control Field, поля данных Data Field, поля контрольной суммы CRC, поля подтверждения ACK Field, поля конца фрейма EOF.

Поле SOF (Start of Frame) данных и удаленного фрейма содержит один доминирующий бит.

Поле арбитража Arbitration Field содержит 11-битовый идентификатор и RTR-бит, показывающий, является данный фрейм фреймом данных или удаленным фреймом. Идентификатор предназначен для адресации сообщений и используется механизмом арбитража.

Управляющее поле Control Field (см. рис. 6.6) содержит 6 битов, из которых 4 бита (DLC0 – DLC3) составляют поле Data Length Code, показывающее количество байтов данных, которое будет передаваться в поле данных; два других бита, зарезервированы для следующих редакций протокола:

Сегмент разграничителя (ACK Delimiter),

Сегмент подтверждения (ACK Slot).

Поле данных Data Field содержит передаваемые данные, причем количество передаваемых байтов указывается в поле Control Field и не может превышать 8.

Поле CRC обеспечивает механизм избыточного контроля по четности передаваемых данных.

Поле подтверждения ACK Field (см. рис. 6.6) содержит участки ACK Slot и ACK Delimiter и выполняет следующую функцию: передающий узел посылает по одному рецессивному биту на каждый участок, а приемник, если он принял сообщение без сбоев, устанавливает на линии доминирующий бит в поле ACK Slot. При наложении рецессивного и доминирующего уровней на линии устанавливается доминирующий, и это событие сигнализирует передающему узлу о том, что передача прошла нормально и повтор не требуется.

Поле конца фрейма EOF содержится в фрейме данных и удаленном фрейме и состоит из семи рецессивных битов. Удаленный фрейм аналогичен по структуре фрейму данных, но не имеет поля данных, а фрейм ошибок и фрейм перегрузки содержат по 2 поля: в первом располагаются флажки ошибок и служебная информация, а второе является полем разграничителя Delimiter и содержит восемь рецессивных битов.

Средства управления доступом к шине в CAN-протоколе. Передающий узел в CAN-протоколе слышит все другие узлы в сети и подтверждают это. Всякий раз, когда шина свободна от передачи, узел может начинать передавать. Если узел передает, эта передача должна быть завершена прежде, чем другой узел попытается передавать. Если два или больше узла начнут передавать в одно и то же время, конфликт решается при помощи неразрушающего (non-destructive) поразрядного алгоритма арбитража, использующего поле арбитража.

Поле арбитража, включенное во все фреймы данных, состоит:

- из 11-битового поля идентификатора,
- RTR-бита.

RTR-бит указывает, является ли фрейм фреймом данных или удаленным фреймом.

11-битовое поле идентификатора передается от старшего значащего бита к младшему. Доминирующий уровень — логический 0. Одновременная передача бита с доминирующим уровнем (логический 0) и бита с рецессивным уровнем (логическая 1) дает в результате уровень логического 0.

В течение передачи поля арбитража каждый передатчик контролирует текущий уровень на шине и сравнивает его с битом, который он должен передавать. Если значения равны, узел способен затем продолжить передачу. Если бит с пассивным уровнем (логическая 1) был передан, а активный бит (логический 0) обнаружен на шине, то данный узел теряет право передачи и должен прекратить передачу последующих данных. Узел, который потерял шину, может сделать попытку передачи снова, когда текущая передача завершена.

Важно следующее: идентификатор CAN с низким значением выигрывает арбитраж.

Из сказанного можно сделать следующие выводы.

Приоритетным является не передающий или приемный узел, а сообщение, имеющее меньшее значение идентификатора. Если в сети один из узлов (сервер) будет ответственным за принятие решений, то он должен иметь наименьший адрес из задействованных.

Вторая возможность, которую дает механизм арбитража, использована в сети верхнего уровня Device Net. В этой сети количество узлов ограничено 64 и для адресации отведены младшие разряды идентификатора, а старшие разряды предназначены для кодирования видов сообщений. Естественно, что сообщение, имеющее 0 в старшем бите, захватит шину первым, независимо от адреса узла приемника. Это, в свою очередь, обеспечивает передачу сообщений первого вида, например об аварии, по сети первыми, независимо от адресов приемных и передающих узлов.

Адресация в CAN-протоколе. CAN – это протокол, ориентированный на использование в условиях помех. Различные сообщения, передающиеся по сети, имеют идентификатор, и каждая станция решает, основываясь на этом идентификаторе, получать или нет это сообщение. Этот идентификатор определен в поле идентификатора CAN-фрейма.

При этом адрес приемника устанавливается в CAN-приемнике путем настройки входных фильтров соответствующих микросхем.

Входные фильтры представляют собой решета, или идентификационные экраны. Любое сообщение, которое проходит через входные фильтры, должно быть обработано процессором обслуживания CAN-контроллера. Чем большее количество единиц может быть отфильтровано, тем меньше нагрузка на процессор.

Микросхемы, поддерживающие CAN-протокол, могут иметь одиночный фильтр или многократные фильтры в зависимости от конкретной реализации.

Существуют следующие два типа входных фильтров:

- фиксированные — фильтры, которые требуют, чтобы биты соответствовали точно один к одному (one-for-one);

- mask-and-match (маскируемые) — фильтры, которые применяют маску к полю идентификатора, прежде чем он сравнивается с приемным регистром кода.

Например, регистр маски сконфигурирован так (табл. 6.2), что полученные биты 10 – 6 идентификатора должны соответствовать битам 10 – 6 в приемном регистре кода. В этом примере биты идентификатора (с 10 по 6) должны быть установлены в 11110, а остальные не имеют значения. Если биты 10 – 6 установлены в 11110, то эти сообщения принимаются независимо от значений битов 5 – 0.

Таблица 6.2

Пример поразрядного маскирования

Фильтрация типа mask-and-match											
10	9	8	7	6	5	4	3	2	1	0	Поразрядное значение идентификатора
1	1	1	1	0	0	0	0	1	1	1	Принятое значение идентификатора
m	m	m	m	m	X	X	X	X	X	X	Регистр маски
1	1	1	1	0	0	0	0	0	0	0	Регистр кода после фильтрации
Принятое сообщение											

Примечание. m – код маски, x – произвольный код (1 или 0).

Управление ошибками. CAN-протокол обеспечивает механизмы обнаружения следующих типов ошибок.

- *Разрядная ошибка* появляется, когда передатчик сравнивает уровень на шине с уровнем, который должен передаваться, и обнаруживает их неравенство. При этом обнаружение активного бита, когда передается пассивный бит, не выдает ошибку в течение передачи поля арбитража, поля ACK Slot или флажка пассивной ошибки.

- *Ошибка подтверждения* возникает, когда передатчик определяет, что сообщение не было подтверждено. Слот подтверждения существует внутри фреймов данных и удаленных фреймов. Внутри этого слота все приемные узлы, независимо от того, являются они пунктом назначения или нет, должны подтвердить получение сообщения.

- *Ошибка заполнения* появляется, когда узел обнаруживает шесть последовательных битов одного и того же значения. В процессе нормальной работы, когда передатчик обнаруживает, что он послал пять последова-

тельных битов одного и того же значения, он заполняет следующий бит противоположным значением (это называется заполнением бита). Все приемники удаляют заполненные биты до вычисления CRC (контрольного кода). Таким образом, когда узел обнаруживает шесть последовательных битов того же значения, возникает ошибка заполнения.

- *CRC-ошибка* появляется, когда CRC-значение (контрольный код) не соответствует значению, сгенерированному передатчиком. Каждый фрейм содержит поле контрольного кода, которое инициализировано передатчиком. Приемники вычисляют CRC и сравнивают его со значением, сгенерированным передатчиком. Если эти два значения не тождественны, то имеет место CRC-ошибка.

- *Ошибка формы* возникает, когда недопустимое разрядное значение обнаружено в области, в которую должно быть передано predetermined значение. В CAN-протоколе существуют некоторые predetermined разрядные значения, которые должны быть переданы в определенных местах. Если недопустимое разрядное значение обнаружено в одной из этих областей, имеет место ошибка формы. CAN позволяет минимизировать негативные последствия наличия дефектного узла в сети при помощи механизма определения состояния узла. Узел может быть в одном из трех состояний ошибки.

- *Ошибка активная* фиксируется, когда активный узел обнаруживает одну из упомянутых ошибок, он передает активный фрейм ошибки, который состоит из шести последовательных доминирующих битов. Эта передача отменит любую другую передачу, проходящую в то же время в CAN, и заставит все другие узлы обнаружить ошибку наполнения, которая, в свою очередь, заставляет их отбрасывать текущий фрейм.

Когда узел в состоянии активной ошибки обнаруживает проблему с передачей, он предотвращает получение всех других данных из пакета сообщений, передавая фрейм активной ошибки. Этот процесс выполняется независимо от того, был ли узел, обнаруживающий ошибку, получателем данных или нет.

- *Ошибка пассивная* фиксируется, когда пассивный узел обнаруживает одну из упомянутых ошибок: он передает фрейм пассивной ошибки, который состоит из шести последовательных пассивных битов. Этот фрейм может быть наложен на передачу, которая ведется в то же время в CAN, при этом данные из передачи не теряются, если другие узлы не обнаруживают ошибку.

- *Шина выключена*: узел на шине в выключенном состоянии и не откликается на любое воздействие на шине. Это логическое отключение от сети.

Общий краткий обзор действий, имеющих в механизме минимизации неисправностей, приведен далее.

- Узлы следят, передают и получают значения счетчиков ошибок.
- Узел начинает передачу в состоянии активной ошибки со счетчиками ошибок, равными нулю (0). Узел в этом состоянии «понимает», что любая обнаруженная ошибка – не неисправность.

- Типы ошибок и точки, в которых они были обнаружены, имеют различный код, который добавляется к текущему общему количеству в зависимости от того, является ли ошибка передаваемой или принимаемой. Значимые величины получения и передачи вызывают декремент этих счетчиков, при этом ноль (0) является минимальным значением. Когда любой из данных счетчиков проходит соответствующий порог, определенный в CAN-протоколе, узел фиксирует пассивное состояние ошибки. В таком состоянии узел полагает, что это — причина ошибки.

- Когда переданное состояние счетчика ошибки в другом узле проходит определенный порог, узел вводит шину в отключенное состояние. Эта спецификация определяет механизмы перехода из состояния отключения шины к состоянию активной ошибки.

- Когда и передающий, и приемный счетчики пассивной ошибки узла декрементируются ниже определенного порога, узел еще раз подтверждает состояние активной ошибки.

Стандартный и расширенный фреймы. CAN-микросхемы поддерживают стандартный или расширенный фрейм. Стандартный фрейм означает, что CAN-микросхема поддерживает 11-битовое поле идентификатора. Расширенный фрейм означает, что микросхема поддерживает 29-битовое поле идентификатора. Новые CAN-микросхемы могут поддерживать форматы как стандартного фрейма, так и форматы расширенного фрейма.

Прерывания в CAN-протоколе. Проектировщики должны учитывать интервал возможных прерываний их CAN-контроллеров при проектировании своих изделий. Так как фрейм данных в CAN-протоколе короткий (от 0 до 8 байт), скорость поступления прерываний на процессор может быть высокой. В связи с этим следует рассматривать CAN как высокоскоростную сеть.

Рис. 6.7 демонстрирует два передаваемых подряд CAN-фрейма данных с минимальным интервалом между фреймами, называемым интервалом межфрейма. Табл. 6.3 показывает CAN-жесткий режим прерывания

для случая, если CAN-приемник получает все фреймы во время текущей связи (непрерывные фреймы в режиме back-to-back).



Рис. 6.7. Минимальный интервал межфрейма

Строка «Количество бит в CAN- фрейме» в табл. 6.3 принимается с условием, что заполнение дополнительными битами отсутствует (естественно, что такое заполнение увеличило бы время между прерываниями).

Таблица 6.3

Трафик прерываний при приеме фреймов в режиме back-to-back

Количество байт данных	0	1	2	3	4	5	6	7	8
Количество бит в CAN-фрейме	47	55	63	71	79	87	95	103	111
Интервал между прерываниями на скорости передачи 125 кбит/с	376	440	504	568	632	696	760	824	888
Интервал между прерываниями на скорости передачи 250 кбит/с	188	220	252	284	316	348	380	412	444
Интервал между прерываниями на скорости передачи 500 кбит/с	94	110	126	142	158	174	190	206	222

Из таблицы видно, что трафик прерываний достаточно интенсивен. На скорости 500 кбит/с прерывания могут происходить каждые 94 мкс при отсутствии информации во фреймах данных. Большинство микроконтроллеров нижнего уровня не может поддерживать такую высокую скорость обработки прерываний. Следовательно, нужно находить компромисс между возможностями CAN-контроллера и его стоимостью. Следует выбирать CAN-контроллер, который обеспечивает соответствующий уровень пред-

варительной фильтрации. Контроллер должен иметь достаточное время для обработки прикладной программы и успевать обслуживать запросы от CAN-сети, или необходимо выделить отдельный микроконтроллер для обслуживания CAN-приемника.

Также следует помнить, что некоторые CAN-микросхемы маскируют только восемь наиболее значащих бит поля идентификатора (не все 11 бит) и имеют один фильтр МАСКИ/СООТВЕТСТВИЯ.

Микросхемы, поддерживающие CAN-протокол

Микросхемы, которые поддерживают CAN-протокол, выпускаются различными поставщиками, такими как Philips, Motorola, Siemens, National Instruments и Intel.

Существуют следующие два типа микросхем.

Встроенные – микросхемы, которые включают в себя CAN-контроллер и один из видов интегрированного микроконтроллера. Это Intel 80196CA, содержащий в одном кристалле стандартный контроллер 80196 и CAN-контроллер 82527; Philips 82C592 и 82C598, имеющие контроллер 80C51 и CAN-контроллер 82C200; Motorola 68HC05X4, 68HC705X4, 68HC705X32 на основе M6805.

Периферийные – микросхемы, которые содержат только CAN-контроллер. Это Intel 82527 с 14 фиксированными входными фильтрами, одним типа Mask-and-Match, и поддержкой стандартного и расширенного фреймов; Philips 82C200 с одним входным фильтром типа Mask-and-Match и поддержкой стандартного фрейма; Siemens SAB 81C90, 81C91 с 16 фиксированными входными фильтрами.

Кроме того, фирмами Philips и Texas Instruments выпускается ряд буферных микросхем, формирующих сигналы CAN-магистрали.

Применение в промышленных приложениях

В настоящее время CAN-протокол активно используется в промышленных сетях. Такие известные фирмы, как Hoheywell и Allan-Bradley, разработали стандарт верхнего уровня SDS и Device Net, причем последний является открытым и на данный момент более 200 фирм выпускают и разрабатывают свои изделия в этом стандарте. Кроме того, достаточно известными в Европе являются стандарты сети верхнего уровня Can Open,

CAL (Германия) и Can Kingdom (Швеция). Все эти сети используют CAN-протокол на физическом и транспортном уровнях. Фирма Advantech выпустила плату PCL-841, имеющую 2 гальванически развязанных CAN-порта на Philips 82C200, и разрабатывает модули удаленного сбора информации с выходом на CAN; фирмы Grayhill и Opto 22 выпустили недорогие периферийные контроллеры, поддерживающие сеть Device Net, в комплекте WAGO I/O System также имеется контроллер с выходом на Can Open, Device Net и CAL.

Фирма Hilscher выпускает богатый набор плат с CAN-протоколами (CAN Open, Device Net, SDS) для распространенных системных шин типа ISA и PCI, для мезонинной шины PC/104, а также в виде OEM-модулей. Для тех изготовителей контроллеров, которые хотят встроить в *jTI* свои изделия для совместимости с CAN-протоколами, не затрачивая время и средства на собственные разработки, ряд отечественных фирм также выпускает изделия с CAN-протоколом, в том числе в популярном формате Micro PC.

Использование CAN-протокола и сетей верхнего уровня на его основе при модернизации отечественных промышленных предприятий позволит разработчикам средств АСУ ТП решить ряд остро стоящих проблем:

- выполнить требования помехоустойчивости;
- обеспечить совместимость с действующими в развитых странах стандартами;
- повысить надежность за счет того, что обеспечивается обязательное подтверждение приема сообщения приемником;
- обеспечить повышение живучести системы при применении режима «мультимастер»;
- снизить стоимость коммуникаций (требуется витая пара).

Программный интерфейс доступа к сети CAN

CHAI (CAN Hardware Abstraction Interface) представляет собой библиотеку, реализующую программный интерфейс доступа к сети CAN на канальном уровне (Data Link Layer) семиуровневой модели ISO/OSI, и не является протоколом высокого уровня (HLP) CAN, таким как CAN open или Device Net.

CHAI поддерживает:

- стандартный и расширенный протокол CAN (11- и 29-битные идентификаторы);
- скорость передачи до 1000 кбод;
- прием CAN-кадров через конфигурируемые приемные очереди FIFO с регистрацией времени приема;
- прием CAN-кадров через конфигурируемые приемные буферы с поддержкой счетчиков числа полученных кадров;
- отправку кадров через очереди FIFO с различными приоритетами;
- автоматическую отправку конфигурируемого ответа на RTR-запрос;
- несколько CAN-контроллеров в одном устройстве;
- одновременную работу нескольких приложений с несколькими CAN-устройствами.

Управление CAN-кадрами

СНАІ предоставляет свои собственные способы управления CAN-кадрами, независимые от типа используемого CAN-контроллера. Полученный из сети CAN-кадр сохраняется либо в приемной очереди, либо в приемном буфере. В очереди сообщения сохраняются последовательно одно за другим по мере их получения, с сохранением последовательности и времени получения (по принципу FIFO). В отличие от очереди, в буфере хранится только последний полученный кадр с определенным идентификатором и общее количество полученных CAN-кадров с этим идентификатором.

Отправляемые CAN-кадры записываются в очередь на отправку, а затем обрабатываются CAN-контроллером. Очередей на отправку может быть несколько с различными приоритетами. CAN-контроллер обрабатывает очереди на отправку в порядке убывания приоритета, то есть сначала в сеть отправляются все CAN-кадры из очереди с наивысшим приоритетом. В дополнение к очереди на отправку сообщение может быть записано в так называемый RTR-буфер. В этом случае оно будет отправлено в сеть только при получении из сети RTR-запроса, идентификатор которого совпадает с идентификатором кадра, записанного в RTR-буфер.

Приемные очереди. Приемные очереди представляют собой наиболее общий способ работы с получаемыми из сети CAN-кадрами. Приложение устанавливает размер очереди (сколько кадров можно сохранить в очереди) и фильтр очереди (определяется по идентификатору кадра: может

ли он быть помещен в эту очередь). К каждому кадру, помещаемому в очередь, добавляется время его получения (timestamp).

Приложение может зарегистрировать функцию, которая будет вызываться библиотекой при наступлении определенного события в очереди и работать с очередью в режиме прерываний. Возможные события:

- получен новый кадр из сети,
- произошла ошибка (Bus-off, error warning limit, аппаратное переполнение контроллера).

Приемные буферы. Приемные буферы создаются по одному на каждый идентификатор, который может быть получен из сети CAN. Каждый приемный буфер хранит последние полученные данные с этим идентификатором. Поэтому если предыдущие данные не были прочитаны приложением к моменту прихода новых, то они затираются. Каждый приемный буфер содержит также счетчик полученных кадров. Этот счетчик предназначен для того, чтобы приложение могло зарегистрировать появление новых данных, периодически проверяя его значение. Работа приложения с приемными буферами в режиме прерывания невозможна, поэтому приемные буферы в основном предназначены для приложений, которые заинтересованы в небольшом количестве сообщений определенного типа и которых интересует лишь последнее полученное значение данных. Примером такого приложения может служить сервер, непосредственно отвечающий на определенное количество типов запросов клиентов. Максимально возможное количество буферов (как приемных, так и RTR) 2048 на каждый CAN-контроллер. Приемные буферы работают только со стандартным форматом кадра (идентификатор 11 бит).

Очереди на отправку. CAN-кадры отправляются приложением в сеть через очередь на отправку. Приложение записывает кадр в очередь, а реальную его отправку в сеть осуществляет CAN-контроллер. Приложение может создать несколько очередей разного размера с различными приоритетами, задающими последовательность, в которой они обрабатываются CAN-контроллером.

RTR-буферы. Если приложению нужно посылать данные в сеть в ответ на RTR-запрос от других узлов, приложение может записать эти данные в RTR-буфер. При получении RTR-запроса, идентификатор которого совпадает с идентификатором буфера, эти данные извлекаются из буфера и отправляются в сеть. Обработка RTR-буферов происходит с наивысшим приоритетом, то есть до обработки любой из очередей на прием или от-

правку. RTR-сообщения могут также обрабатываться обычным образом, через приемные очереди. Максимально возможное количество буферов (как приемных, так и RTR) 2048 на каждый CAN-контроллер. RTR-буферы работают только со стандартным форматом кадра (идентификатор 11 бит).

Описание API. Каждый контроллер идентифицируется целым числом (каналом ввода-вывода) начиная с 0. Изначально с каждым контроллером ассоциирована одна очередь на передачу (номер очереди 0) с максимальным приоритетом и одна очередь на прием (номер очереди 0), фильтр которой настроен на прием всех возможных кадров. Другие очереди могут быть созданы впоследствии. Существует глобальный аппаратный фильтр, относящийся к CAN-контроллеру в целом, этот фильтр является частью CAN-контроллера.

При получении кадра из сети CAN в первую очередь проверяет наличие буфера на прием или RTR-буфера (если получен RTR-кадр). Если буфер обнаружен, то содержимое кадра кладется в буфер (в случае приемного буфера) либо в сеть отправляется содержимое буфера (в случае RTR-буфера). Если буфер не обнаружен, CAN последовательно перебирает очереди на прием в порядке возрастания номера очереди и сравнивает идентификатор полученного кадра с фильтром очереди. Если фильтр позволяет, то CAN помещает сообщение в эту очередь. Если идентификатор кадра не проходит проверку ни с одним фильтром, кадр сбрасывается.

Отправка кадров: при получении запроса на передачу CAN последовательно проверяет очереди в порядке убывания приоритета (возрастания номера очереди). Если текущая очередь не пуста, из нее извлекается один кадр и отправляется в сеть, затем процедура сканирования очередей повторяется с начала (начиная с очереди с наивысшим приоритетом, номер очереди 0).

Программный интерфейс библиотеки CAN состоит из пяти групп функций:

- Базовые функции, обязательно присутствующие во всех версиях CAN. Эти функции позволяют конфигурировать CAN-контроллер и отправлять/принимать CAN-кадры через две predetermined очереди: одна очередь – на прием, другая – на отправку.

- Функции, позволяющие создавать очереди на прием и отправку CAN-кадров, дополнительные к двум predetermined очередям. Эти

функции могут отсутствовать в редакции библиотеки для микроконтроллеров из-за ограничений на производительность и память.

- Функции, предназначенные для работы с буферами (приемными и RTR). Эти функции могут отсутствовать в редакции библиотеки для микроконтроллеров из-за ограничений на производительность и память.

- Статусные функции предназначены для определения состояния CAN-контроллера и CAN-адаптера. Эти функции могут отсутствовать в редакции библиотеки для микроконтроллеров из-за ограничений на производительность и память.

- Дополнительные функции, зависящие от типа применяемого CAN-контроллера и платформы, могут быть непереносимы с одной платформы на другую. Примером таких функций является функция включения/выключения специфичного для CAN-контроллера Phillips SJA1000 режима Listen Only Mode.

Структура данных для представления CAN-кадра имеет следующий прототип:

```
typedef struct {
    _u32 id;          /* идентификатор кадра */
    _u8 data[8];     /* данные */
    _u8 len;         /* фактическая длина поля данных, от 0 до 8 байт */
    _u16 flags;      /* bit 0 - RTR, bit 1 - HOVR, bit 2 - EFF, bit 3 - SOVR */
    _u32 ts;        /* отметка времени получения (timestamp) в микросекундах */
} canmsg_t;
```

Примечания:

- если бит 0 поля `flags` выставлен в 1 – кадр помечен как RTR;
- если бит 1 поля `flags` выставлен в 1 – при приеме кадра произошло аппаратное переполнение CAN-контроллера;
- если бит 2 поля `flags` выставлен в 1 – кадр помечен как EFF (Extended Frame Format, идентификатор 29 бит);
- если бит 3 поля `flags` выставлен в 1 – при приеме кадра произошло программное переполнение одной из приемных очередей;
- время, указываемое в поле t_s , измеряется от момента открытия канала в микросекундах. Переполнение этого поля наступает примерно через 71 мин после открытия канала. После переполнения время вновь отсчитывается от нуля.

Для работы с типом данных `canmsg_t` определены следующие функции:

- `void msg_zero (canmsg_t *msg);` – обнуляет кадр `msg`. После вызова кадр `msg` представляет собой кадр стандартного формата (SFF – standart frame format, идентификатор 11 бит), с длиной поля данных ноль, данные и все остальные поля выставлены в ноль.

- `_s16 msg_isrtr (canmsg_t *msg);` – возвращает не ноль (`true`), если `msg` – RTR-кадр.

- `void msg_setrtr (canmsg_t *msg);` – помечает `msg` как RTR-кадр.

- `_s16 msg_iseff (canmsg_t *msg);` – возвращает не ноль (`true`), если `msg` – кадр расширенного формата (EFF – extended frame format, идентификатор 29 бит).

- `void msg_seteff (canmsg_t *msg);` – помечает `msg` как кадр расширенного формата.

- `_s16 msg_ishovr (canmsg_t *msg);` – возвращает не ноль (`true`), если в поле `flags` кадра `msg` выставлен бит HOVR (hardware overrun), это означает, что во время приема кадра `msg` произошло аппаратное переполнение CAN-контроллера;

- `_s16 msg_issovr (canmsg_t *msg);` – возвращает не ноль (`true`), если в поле `flags` кадра `msg` выставлен бит SOVR (software overrun), это означает, что во время приема кадра `msg` произошло программное переполнение одной из приемных очередей.

Для работы с RTR и приемными буферами используется одинаковая структура данных `canbuf_t`:

```
typedef struct {
    _u8 data[8];    /* данные */
    _u8 len;        /* фактическая длина поля данных, от 0 до 8 байт */
    _u32 cnt;       /* счетчик полученных кадров, только для приемных буферов */
} canbuf_t;
```

В целях переносимости в заголовочном файле `chai.h` определены следующие типы данных:

- `_u8` – беззнаковое целое длины 8 бит (1 байт), `u16` – длины 16 бит (2 байта), `u32` – длины 32 бит (4 байта);

- `_s8` – знаковое целое длины 8 бит (1 байт), `s16` – длины 16 бит (2 байта), `s32` – длины 32 бит (4 байта).

Список функций, обязательно присутствующих во всех версиях CHAI:

- `_s16 CiInit(void);`
- `_s16 CiOpen(_u8 chan, _u8 flags);`
- `_s16 CiClose(_u8 chan);`
- `_s16 CiStart(_u8 chan);`
- `_s16 CiStop(_u8 chan);`
- `_s16 CiSetFilter(_u8 chan, _u32 acode, _u32 amask);`
- `_s16 CiSetBaud(_u8 chan, _u8 bt0, _u8 bt1);`
- `_s16 CiWrite(_u8 chan, canmsg_t *mbuf, _s16 cnt);`
- `_s16 CiRead(_u8 chan, canmsg_t *mbuf, _s16 cnt);`
- `_s16 CiSetCB(_u8 chan, _u8 ev, void (*ci_handler) (_s16)).`

Device Net

Device Net предназначен для объединения промышленных устройств в единую сеть с общей шиной. Device Net – сетевое решение, которое предоставляет полноценную инфраструктуру для установления соединений между устройствами с поддержкой полной взаимозаменяемости устройств разных производителей. Device Net – открытый стандарт, и производители не обязаны приобретать лицензии для разработки Device Net-совместимых устройств.

Device Net – это протокол, ориентированный на соединения. Прежде чем между двумя узлами сети могли быть переданы какие-либо данные, должно быть установлено сетевое соединение. Для установления сетевых соединений узел использует либо Unconnected Message Manager (UCMM – менеджер сообщений без установления соединения), либо Unconnected Port. Для обмена данными в Device Net используются два основных типа сообщений: явные сообщения (explicit messages) и сообщения ввода-вывода (I/O messages).

Явные сообщения представляют собой обмен запрос – ответ, имеющий своей целью обеспечение служебных функций, например, таких как удаленная конфигурация узла и настройка параметров соединения. Сообщения ввода-вывода предназначены для быстрого обмена данными. Сообщения обоих типов могут обрабатывать данные любой длины благодаря поддерживаемому механизму фрагментации (без фрагментации максимальная длина сообщения 7 байт).

Device Net использует поле идентификатора CAN-кадра для поддержки схемы адресации в сети. Device Net поддерживает до 64 узлов. Все идентификаторы поделены на группы (рис. 6.8).

Группа 1 – содержит наиболее приоритетные идентификаторы. Всего 1024 идентификатора, то есть по 16 на каждый узел.

Группа 2 – отличается внутренним использованием битов. Она введена для поддержки CAN контроллеров, которые фильтруют поступающие сообщения только по первым восьми битам. Это так называемые Basic-CAN-контроллеры. Некоторые идентификаторы 2-й группы используются системой в специальных целях.

Группа 3 – предоставляет для каждого узла по 7 низкоприоритетных идентификаторов.

10	9	8	7	6	5	4	3	2	1	0	
0	Group 1 Message ID				Source MAC ID						Группа 1
1	0	MAC ID					Group 2 Message ID				Группа 2
1	1	Group 3 Message ID			Source MAC ID						Группа 3
1	1	1	1	1	Group 4 Message ID						Группа 4

Рис. 6.8. Поле идентификатора в Device Net

Группа 4 – используется для обработки внештатных ситуаций, таких, например, как восстановление работоспособности узла после сбоя.

Для адресации каждому узлу в сети Device Net назначается так называемый MAC ID (Media Access Control Identification – идентификатор контроля доступа к сети). Размер поля MAC ID составляет 6 бит, поэтому в сети Device Net может находиться до 64 узлов. Уникальность каждого MAC ID обеспечивается процедурой проверки его уникальности, которую проходит каждый узел сети во время подключения. Если узел при подключении к сети обнаруживает, что в сети уже работает узел с таким же MAC ID, то он регистрирует ошибку и отключается от сети.

Узел сети Device Net может быть клиентом или сервером. Сервер или клиент могут и отправлять сообщения (producer), и принимать сообщения (consumer), или одновременно и отправлять и принимать. Следует отметить, что с точки зрения схемы Master/Slave Master является клиен-

том, а Slave – сервером. Обычно клиент посылает (produce) какое-либо сообщение и принимает (consume) ответ на него. Сервер обычно принимает (consume) запрос и посылает (produce) ответ на него. Device Net имеет несколько вариаций этой стандартной модели. Некоторые соединения в клиенте или сервере могут только принимать сообщения. Эти соединения представляют собой конечные пункты доставки сообщений типа Cyclic или Change-of-State. А соединения, используемые только для передачи, представляют собой источники сообщений типа Cyclic или Change-of-State.

Объектная модель. Для описания набора доступных сетевых сервисов, видимого снаружи поведения узлов сети, а также общих способов доступа к информации внутри узла и обмена этой информацией между узлами Device Net использует абстрактную объектную модель. Каждый узел сети Device Net описывается как набор объектов (рис. 6.9). Каждый объект узла сети имеет строго определенные спецификацией атрибуты, которые задают его свойства и сервисы (службы), определяющие его поведение. Взаимодействие между объектами внутри узла и взаимодействие объектов друг с другом через сеть происходит посредством изменения этих атрибутов и обмена запросами к сервисам. Поясим назначение отдельных объектов в схеме на рис. 6.9.

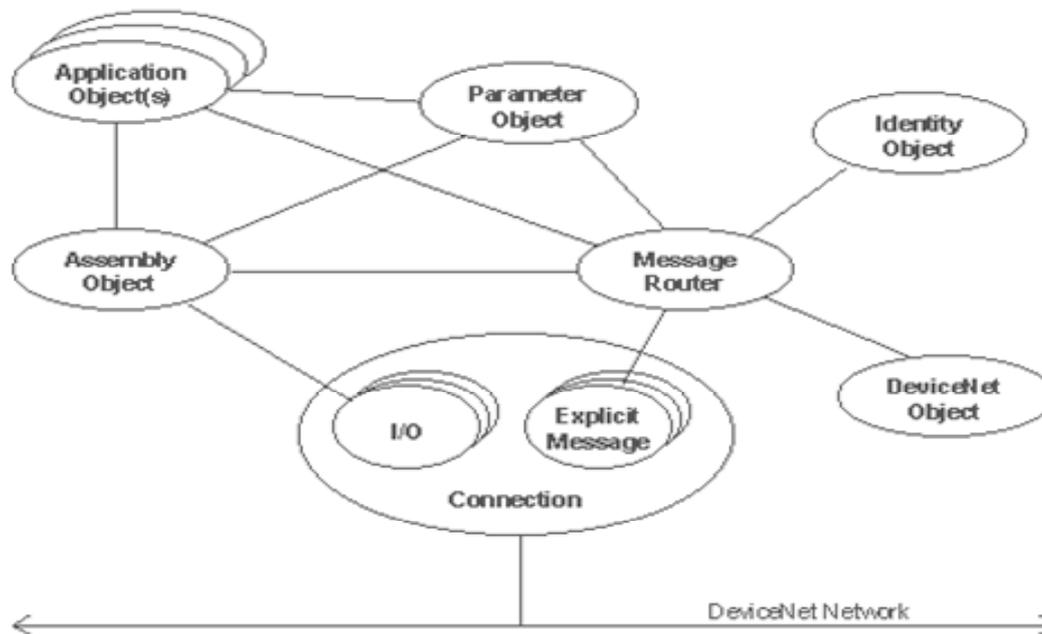


Рис. 6.9. Узел сети Device Net

- *Connection Object* – обеспечивает взаимодействие данного узла с другими узлами по сети Device Net. Каждый узел имеет как минимум два таких объекта. Каждый такой объект представляет одну из конечных точек

соединения между двумя узлами. Два объекта соответствуют двум типам соединения: Explicit Messaging and I/O Messaging. Explicit-сообщения содержат адрес атрибута, значение атрибута и код сервиса, который необходимо осуществить. I/O-сообщения содержат только данные. Вся информация о том, что необходимо делать с этими данными содержится в Connection-объекте, ассоциированном с этим I/O-соединением.

- *Assembly Object* – группирует данные от разных объектов данного узла в один блок данных для последующей передачи как целого.

- *Application Object* – задает специфику поведения данного устройства (содержит специфичный для данного узла код приложения).

- *Device Net Object* – определяет конфигурацию и статус физического соединения с сетью Device Net. Содержит следующие атрибуты: адрес узла (MAC ID), baudrate, обработчик события Bus-Off, Bus-Off счетчик и т.п.

- *Identity Object* – содержит такие данные об устройстве, как VendorID, тип устройства, Product code, версия, статус, серийный номер, название устройства.

- *Message Router* – направляет приходящие Explicit-сообщения объектам, которым эти сообщения были адресованы. Этот объект является внутренним для устройства и не виден со стороны сети.

- *Parameter Object* – содержит основные данные о конфигурации устройства и предоставляет средствам конфигурации возможность удаленно оперировать устройством.

Предопределенный набор соединений Master/Slave (Predefined Master/Slave Connection Set).

В целом Device Net предполагает динамическое установление и конфигурирование соединений между устройствами. Однако большинство устройств имеют детерминированное и жестко заданное поведение. Расходовать дополнительные вычислительные ресурсы на динамическое конфигурирование соединений в таких устройствах нецелесообразно. Предопределенный набор соединений Master/Slave предназначен как раз для таких устройств, состояние соединений которых определено раз и навсегда, а конфигурирование соединений происходит при включении питания. Все что нужно Master-устройству для начала работы с такими Slave-устройствами – это объявить себя Master. Предопределенный набор соединений Master/Slave состоит из одного explicit-соединения, которое позволяет устанавливать несколько типов I/O-соединений:

- *Bit-Strobe-соединения* используются для быстрого обмена I/O-данными между Master и его Slave. Bit-Strobe Command посылает один бит информации каждому Slave, чей MAC ID находится в скан-списке Master. Bit-Strobe Command содержит битовую строку длиной 64 бит (8 байт), по одному биту на каждый MAC ID сети Device Net. Bit-Strobe Response воз-

вращает Master до 8 байт данных (один CAN-frame) от каждого запрошенного Slave.

- *Poll-соединения* используются для обмена I/O-данными любой длины между Master и его Slave (данные отсылаются по запросу с использованием механизма фрагментации).

- *Change of State/Cyclic-соединения* для обмена I/O-данными любой длины между Master и его Slave с использованием Change of State и Cyclic-способов передачи данных (данные отсылаются или при изменении состояния устройства, или периодически).

Профили устройств. Профили устройств используются в Device Net для того, чтобы устройства одного типа разных производителей были взаимозаменяемы. Спецификация Device Net описывает множество типовых профилей устройств. Профиль устройства содержит: определение объектной модели устройства, какие объекты и в каком количестве содержит устройство, спецификацию протокола Device Net, содержание спецификации Device Net.

6.2. Основы PROFIBUS (PROcess FieLd BUS)

По сравнению с обычной реализацией структур автоматизации, уже на первый взгляд видны преимущества применения последовательной полевой шины.

Экономия средств здесь образуется благодаря меньшей стоимости кабеля (кабель имеет меньшую длину) и применению полевых приборов. Это возможно, однако, только при стандартизованной и открытой полевой шине.

В 1987 году для немецкой промышленности был разработан и принят стандарт DIN E 19245 PROFIBUS. В 1996 году этот стандарт стал международной нормой EN 50170.

Модель ISO/OSI

Архитектура протоколов PROFIBUS ориентирована на уже установленные национальные и международные нормы. Так, архитектура протоколов базируется на модели OSI (Open System Interconnection).

Модель ISO/OSI для коммуникационных стандартов состоит из 7 уровней, подразделяющихся на два класса: ориентированных на пользователя с уровня 5 по уровень 7 и ориентированных на сеть (уровни 1 – 4).

Уровни с 1 по 4 описывают пересылку передаваемых данных из одного пункта в другой, в то время как уровни с 5 по 7 предоставляют в распоряжение пользователя доступ к сети в соответствующей форме.

Архитектура протоколов и профили

Из рис. 6.10 представляющего архитектуру протоколов PROFIBUS, можно видеть, что в ней реализованы уровни 1, 2 и 7. Для уровней 1 и 2 принят стандарт США EIA (Electronic Industries Association) RS485, международные нормы IEC 870-5-1 (Telecontrol Equipment and System) и EN 60870-5-1. Метод доступа к сети, службы передачи и управления данными ориентируются на DIN 19241, части 1 – 3 и нормы IEC 955 Process Data Highway/Тип С. Функции управления (FMA7) ориентированы на ISO DIS 7498-4 (Management Framework). С точки зрения пользователя PROFIBUS подразделяется на 3 профиля протокола: DP, FMS и PA.

PROFIBUS-DP применяет уровни 1 и 2, а также пользовательский интерфейс. Уровни с 3 по 7 не используются. Благодаря такой архитектуре достигается быстрая передача данных. Direct Data Link Mapper (DDLМ) организует доступ к уровню 2. В основу пользовательского интерфейса положены необходимые пользовательские функции, а также системные и аппаратно-зависимые функции различных типов PROFIBUS-DP-приборов.

Этот профиль протокола PROFIBUS оптимизирован для быстрого обмена данными специально для коммуникаций между системами автоматизации и децентрализованной периферией на полевом уровне.

	PNO - профиль для DP устройств	PNO - профиль для EMS устройств	PNO - профиль для DP устройств
	Основные функции Расширенные функции		Основные функции Расширенные функции
	DP user Interface Direct Data Link Mapper (DDLМ)	Application Layer Interface (ALI)	DP user Interface Direct Data Link Mapper (DDLМ)
Layer 7 Application	↑	Application Layer Fieldbus Message Specification (FMS)	↑
Layer 3-6	Не используется		
Layer 2 Link	↓	Data Link Layer Fieldbus Data Link (FDL)	↓
Layer 1 Phisic		Data Link Layer Fieldbus Data Link (FDL)	IEC interface
	Physical layer RS485/LWL	Physical layer RS485/LWL	IEC 1158-2

Рис. 6.10. Архитектура протоколов PROFIBUS

В **PROFIBUS-FMS** применяются уровни 1, 2 и 7. Пользовательский уровень состоит из FMS (Fieldbus Message Specification) и LLI (Lower Layer

Interface). FMS содержит пользовательский протокол и предоставляет в распоряжение коммуникационные службы.

LLI реализует различные коммуникационные связи и создает для FMS аппаратно-независимый доступ к уровню 2.

FMS применяется для обмена данными на уровне ячеек (PLC и PC). Мощные FMS-сервисы открывают широкие области использования и большую гибкость при передаче больших объемов данных.

PROIBUS-DP и PROFIBUS-FMS применяют одинаковую технику передачи и единый протокол доступа к шине и поэтому могут работать через общий кабель.

PROFIBUS-PA применяет расширенный PROFIBUS-DP-протокол передачи данных. Техника передачи согласно IEC 1158-2 обеспечивает надежность и питание полевых приборов через шину. Приборы PROFIBUS-PA могут благодаря применению специальных устройств (PROFIBUS-PA-Links) в простейшем случае интегрироваться в PROFIBUS-DP-сеть.

PROFIBUS-PA – специальная концепция, позволяющая подключать к общей шине датчики и приводы, находящиеся во взрывоопасной зоне.

Уровни PROFIBUS

Физический уровень (Layer 1) для DP/FMS (RS-485)

В основной версии для экранированной витой пары уровню 1 PROFIBUS соответствует симметричная передача данных по стандарту EIA RS-485 (также обозначается H2). Проводники шинных сегментов замкнуты с обеих сторон, скручены и экранированы (рис. 6.11).

Способ передачи. Для PROFIBUS назначен способ передачи RS-485, базирующийся на полудуплексной, асинхронной синхронизации. Данные передаются внутри 11-разрядного кадра (рис. 6.12) в NRZ-коде (Non Return to Zero). Значения сигнала (биты) не изменяются во время передачи сигнала. В то время как передача бинарного значения “1” соответствует положительному значению на проводнике RxD/TxD-P (Receive/Transmit-Data-P), напротив, на проводнике RxD/TxD-N (Receive/Transmit-Data-N) присутствует “0”. Состоянию покоя между отдельными телеграммами соответствует двоичный сигнал “1” (рис. 6.13).

В литературе часто также оба проводника PROFIBUS обозначают как А-проводник и В-проводник. При этом А-проводник соответствует RxD/TxD-N, а В-проводник – RxD/TxD-P.

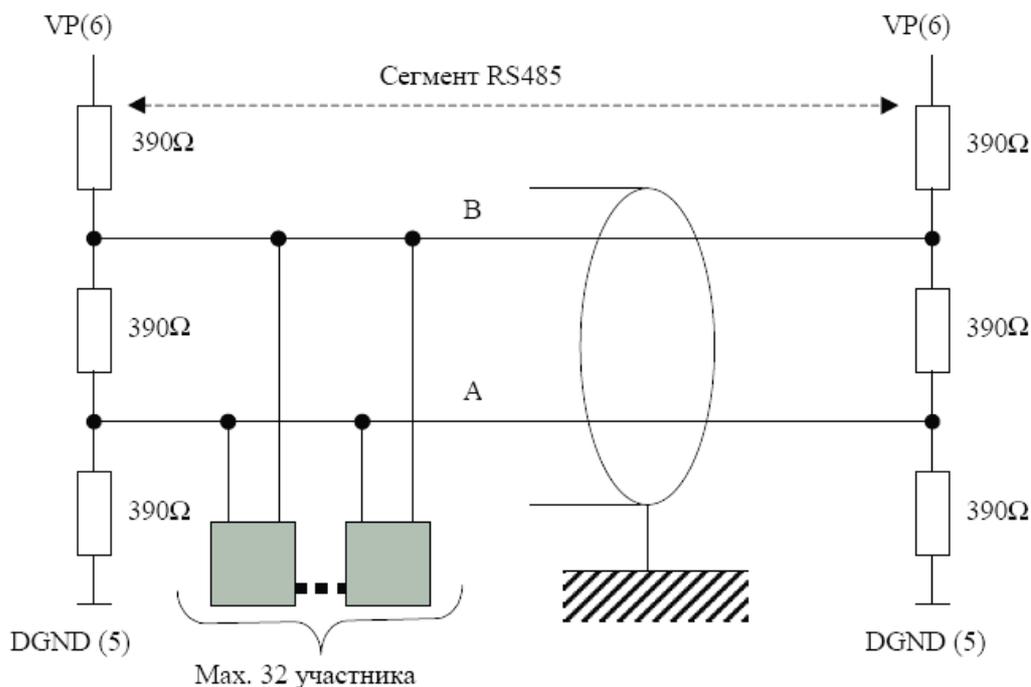


Рис. 6.11. Структура шинного сегмента RS-485

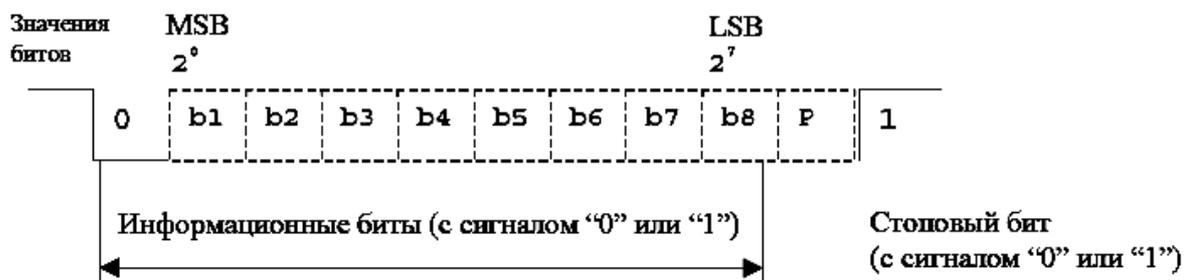


Рис. 6.12. PROFIBUS UART-кадр:
 LSB – Least Signification Bit (младший бит);
 MSB – Most Signification Bit (старший бит)

Шина. В табл. 6.4 приведена максимально допустимая длина провода (длина сегмента) системы PROFIBUS. Эта длина зависит от скорости передачи. Внутри сегмента может быть до 32 участников.

Таблица 6.4

Максимальная длина сегмента в зависимости от скорости

Скорость передачи, кбит/с	9,6 – 187,5	500	1500	12000
Длина сегмента, м	1000	400	200	100

Данные о максимальной длине сегмента в табл. 6.4. взяты из норм PROFIBUS. В табл. 6.5 приведены параметры кабеля типа А.

Таблица 6.5

Спецификации PROFIBUS-кабеля типа А.

Параметр	Характеристика
Волновое сопротивление	От 135 до 165 Ом при частотах измерения от 3 до 20 МГц
Погонная емкость	< 30 пФ/м
Площадь сечения	> 0,34 мм ² , соответствует AWG22
Тип кабеля	Витая пара, 1×2 или 2×2, или 1×4 провода
Погонное сопротивление	< 110 Ом/км
Затухание сигнала	Мах 9 дБ на всей длине отрезка провода
Экранирование	Медная оплетка или оплетка и экран из фольги

Подключение шины. В качестве стандарта для подключения участников к шине в нормах PROFIBUS EN 50170 рекомендуется 9-штырьковый штекер, который изображен в табл. 6.6. У каждого участника есть такой разъем с бухтовыми контактами, шинный кабель имеет разъем со штырьковыми контактами.

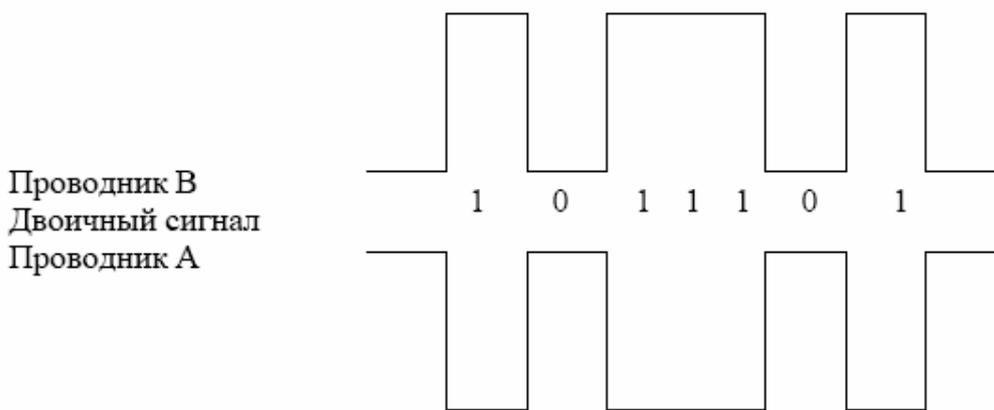
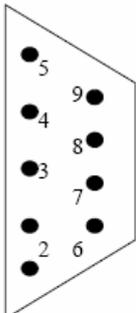


Рис. 6.13. Структура сигнала при передаче NRZ-кодом

Расположение контактов

Вид	Номер контакта	Название сигнала	Обозначение
	1	SHIELD	Экран, например земля
	2	M24	-24 В
	3	RxD/TxD-P	Прием/передача данных, плюс, провод В
	4	CNTR-P	Сигнал для управления направлением передачи
	5	DGND	Данные
	6	VP	Напряжение питания, плюс
	7	P24	+24 В
	8	RxD/TxD-N	Прием/передача данных, минус, провод А
	9	CNTR-N	Сигнал для управления направлением передачи, минус

Окончание шины. Шинные провода данных с обеих сторон замкнуты на согласованные нагрузки (см. рис. 6.11). Благодаря этим сопротивлениям устанавливается безопасный потенциал покоя на проводах шины, когда участники не обмениваются сообщениями (потенциал покоя между телеграммами). Шинные нагрузки имеются почти во всех стандартных разъемах PROFIBUS и могут быть активизированы с помощью переключателей.

Если используется шина со скоростью передачи более 1500 кбит/с, то нужно на основании потребляемой мощности подключенных участников и отраженной мощности использовать шинный штекер с дополнительной индуктивностью (рис. 6.14).

Физический уровень (Layer 1) для DP/FMS (световоды)

Дальнейшее использование уровня 1 PROFIBUS по норме PNO (Profibus Nutzer Organisation) “Техника оптической передачи для PROFIBUS” – это передача данных с помощью световодов. Благодаря оптоволокну внутри установки PROFIBUS между участниками достигается

расстояние до 15 километров. Световодная техника устойчива к электромагнитным помехам и устанавливает безопасную разность потенциалов между участниками. Благодаря простоте подключения световодов, специальным пластиковым световодам, эта технология пришла на полевой уровень.

Среда передачи. В качестве среды передачи используются световоды со стеклянными или пластиковыми волокнами. В зависимости от используемого типа проводника длина связи может быть до 15 километров при стеклянных световодах и до 80 метров при пластиковых.

Подключение шины. Для подключения участников к световоду имеется различная техника (различные модули)

- **Модуль OLM (Optical Link Module).** Похож на репитер RS-485. Имеет два функционально разделенных электрических канала и выходы для одного или двух оптических каналов. Модули OLM соединяются с отдельными участниками или сегментами шины через интерфейс RS-485 (рис. 6.15).

- **Модули OLP (Optic Link Plug).** С помощью модулей OLP можно соединять друг с другом оптическим волокном пассивных участников (Slave). Модули OLP подключаются прямо на 9-штырьковый штекер участника. OLP получает энергию от участника и поэтому не нуждается в напряжении питания. Как видно из рис. 6.16, для подключения активных участников шины (Master) к OLP-кольцу всегда используется OLM.

- **Интегрированное LWL-подключение** (от нем. Licht Wellen Leiter). Прямое подключение участников PROFIBUS к световоду. Возможно у приборов со встроенным LWL-вводом.

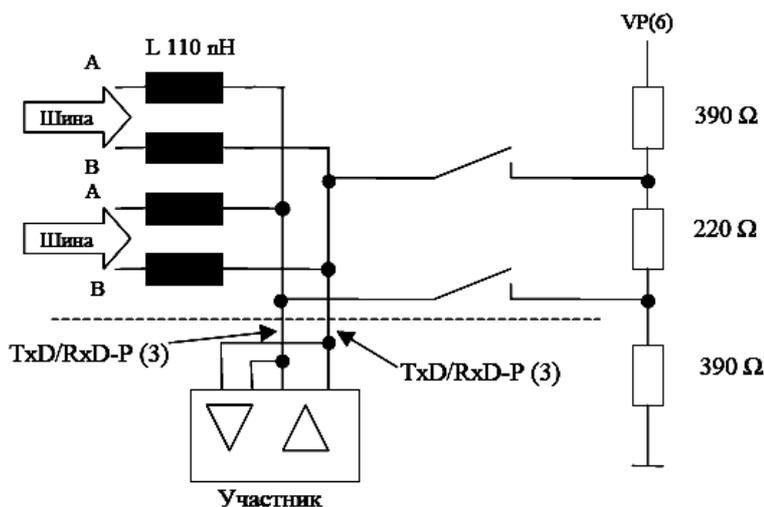


Рис. 6.14. Схема шинного штекера для скорости передачи более 1500 кбит/с

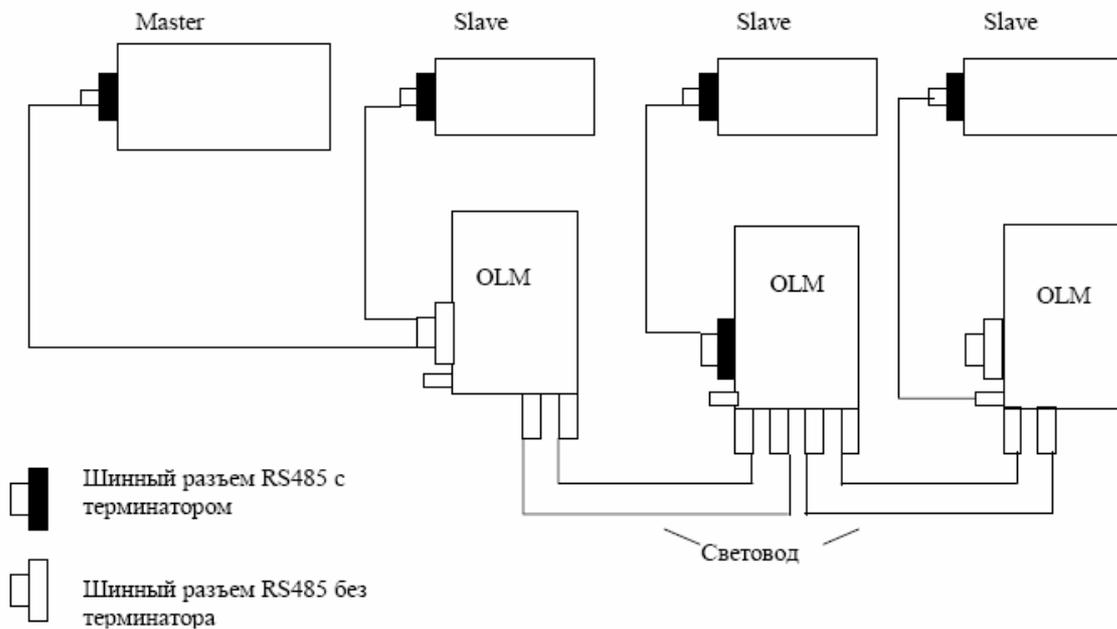


Рис 6.15. Пример шинной конфигурации с OLM-технологией:

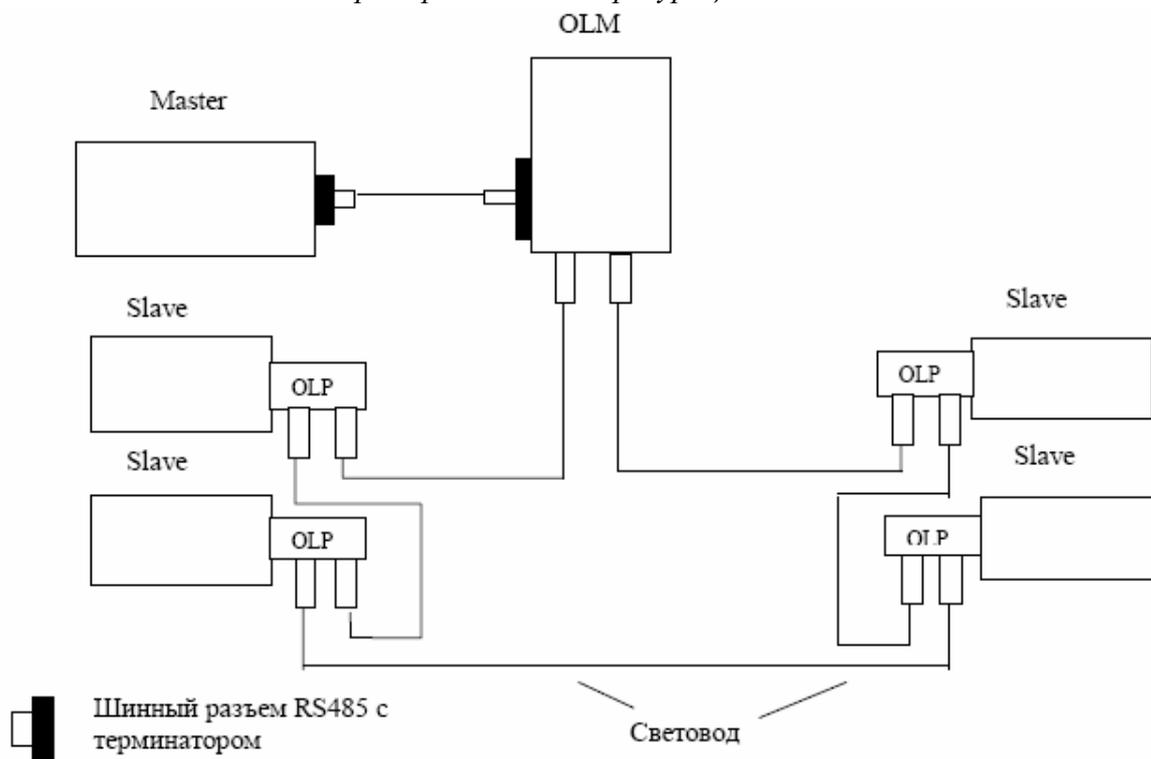


Рис 6.16. Оптическое однопроводное кольцо с OLM-технологией:

Физический уровень (Layer 1) для PA

В PROFIBUS-PA используется передающая техника по IEC 1158-2. Эта техника позволяет достигнуть электробезопасности и питания полевых приборов прямо через шину. Для передачи данных используется бит-синхронизированный с манчестерским кодом протокол передачи без постоянной составляющей (обозначается также как H1). При передаче данных с помощью манчестерского кода бинарный "0" передается как смена фронта с 0 на 1, а бинарная "1" – как смена фронта с 1 на 0. Данные передаются с помощью модуляции +/-9 мА основного тока шинной системы I_b (рис. 6.17).

Скорость передачи составляет 31,25 кбит/с. В качестве среды передачи используется витой экранированный или неэкранированный провод. Шина, как это видно из рис. 6.18, состоит из сегментов, к которым подключены участники, сегменты замкнуты на RC-цепочки. К сегменту шины PA может быть подключено максимум 32 участника. Максимальная длина сегмента сильно зависит от применяемого источника питания, типа провода и потребления тока подключенными участниками.

Шинный провод. В качестве среды передачи для PROFIBUS-PA применяется 2-жильный кабель, технические

данные которого не установлены/не нормированы (табл. 6.7). Свойства типов кабелей определяют максимальную длину шины, число подключаемых участников и чувствительность к электромагнитным шумам. На основании этого установлены для стандартных типов кабелей электрические и механические свойства.

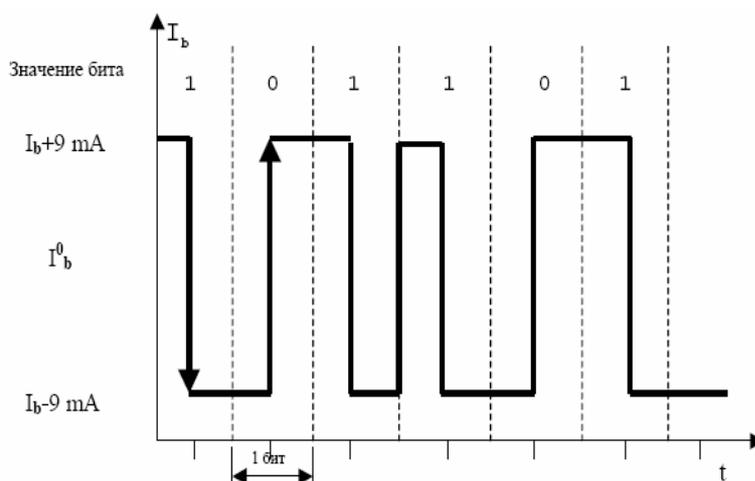


Рис. 6.17. Передача данных в PROFIBUS-PA с помощью модуляции тока (Манчестерский код II)

В DIN 61158-2 предложены для применения стандартные кабели для PROFIBUS-PA, называемые типами А...D.

Таблица 6.7

Предлагаемые типы кабеля для PROFIBUS-PA

Структура кабеля	Тип А	Тип В	Тип С	Тип D
	Витой, двухжиль- ный, экра- нирован- ный		Одна или не- сколько витых пар, экраниро- ванный	Несколько витых пар, неэкрани- рованный
Площадь сечения (номинальная), мм ²	0,8 (AWG18)	0,32 (AWG22)	0,13 (AWG26)	1,26 (AWG16)
Погонное сопро- тивление (посто- янный ток), Ом/км	44	112	264	40
Волновое сопро- тивление при 31,25 кГц, Ом	100	100	–	–
Затухание при 39 кГц, дБ/км	3	5	8	8
Емкостное рассо- гласование, нФ/км	2	2	–	–
Групповое время запаздывания (7,9 – 39 кГУ), с/км	1,7	–	–	–

Структура кабеля	Тип А	Тип В	Тип С	Тип D
	Витой, двухжильный, экранированный	Одна или несколько витых пар, экранированный	Несколько витых пар, неэкранированный	Несколько невитых пар, неэкранированный
Степень экранирования, %	90	–	–	
Рекомендуемая длина сети, включая волновые согласующие шлейфы, м	1900	1200	400	200

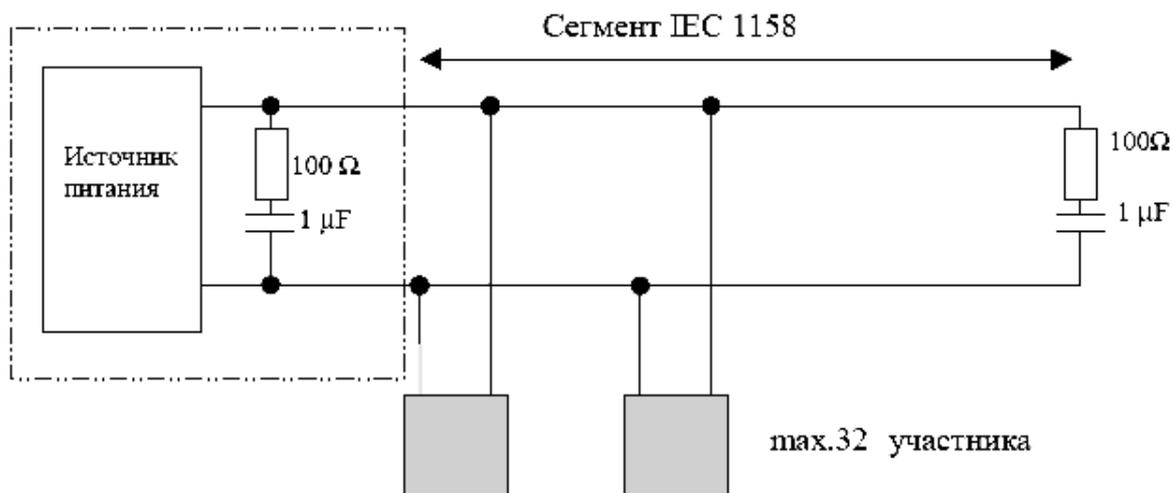


Рис. 6.18. Структура шинного сегмента PA

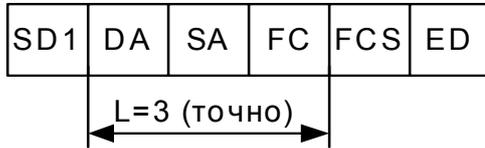
Fieldbus Data Link (Layer 2)

Согласно модели OSI на втором уровне реализуются функции управления доступом к шине, обеспечение безопасности данных, а также выполнение протокола передачи и формирование телеграмм. Уровень 2 обозначается в PROFIBUS как FDL-уровень (Fieldbus Data Link).

Формат телеграммы уровня 2 (рис. 6.19) способствует большей безопасности передачи. Вызывающая телеграмма имеет расстояние Хэмминга HD (Hamming Distance), равное 4. При HD = 4 может быть распознано до 3 одновременных ошибок в фальсифицированных битах телеграммы дан-

ных. Это достигается благодаря применению особых, стартового и завершающего, знаков телеграммы, постоянно скользящей синхронизации, биту четности и контрольному байту.

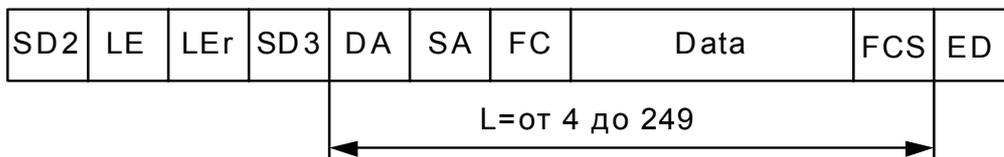
Формат с постоянной длиной информационного поля



Формат с постоянной длиной информационного блока с данными



Формат с переменной длиной информационного блока



Короткое квитирование



Телеграмма - токен (маркер)



Рис. 6.19. Формат PROFIBUS-телеграмм:

L – длина информационного поля; SC (Single Character) – отдельный символ, используется только для квитирования ($SC = E5h$); $SD1$ - $SD4$ (Start Delimiter) – стартовый байт для отличия различных форматов телеграмм ($SD1 = 10h$, $SD2 = 68h$, $SD3 = A2h$, $SD4 = DCh$); LE / LEr (LEngth) – байт длины, указывает длину информационных полей у телеграмм с переменной длиной; DA (Destination Adress) – байт адреса цели, содержит информацию о приемнике; SA (Source Adress) – байт адреса источника, содержит информацию о передатчике; FC (Frame Control) – контрольный байт содержит информацию о службе для данного сообщения и приоритет сообщения; $Data Unit$ – поле данных, может также содержать возможные расширения адреса телеграммы пользовательских данных; FCS (Frame Check Sequence) – проверочный байт, содержит контрольную сумму телеграммы, которая образуется операцией “И” без бита переполнения; ED (End Delimiter) – окончательный байт, указывает на конец телеграммы ($ED=16h$)

При этом могут быть распознаны следующие ошибки:

- ошибка символьного формата (четность, переполнение, ошибка фрейма);
- ошибки протокола;
- ошибки разделителей начала и окончания;
- ошибки байта проверки фрейма;
- ошибки длины телеграммы.

Телеграмма, у которой распознана ошибка, повторяется, по крайней мере, один раз.

Имеется возможность повторять телеграммы, проходящие по уровню 2 до 8 раз (шинный параметр “Retry”). Уровень 2 может осуществлять наряду с передачей данных “точка к точке” также коммуникации во многие точки – Broadcast и Multicast.

При коммуникациях Broadcast активный участник посылает сообщение всем остальным участникам (Master и Slave). Прием данных не квитируется.

При коммуникациях Multicast активный участник посылает сообщение группе участников (Master или Slave). Прием данных не квитируется.

Службы, предлагаемые уровнем 2, приведены в табл. 6.8.

Таблица 6.8

Службы передачи PROFIBUS

Служба	Функции	DP	PA	FMS
SDA	Посылка данных с квитиowaniem			x
SRD	Данные посылаются и принимаются с квитиowaniem	x	x	x
SDN	Данные посылаются без квитиования	x	x	x
CSRД	Циклическая посылка и прием с квитиowaniem			x

В PROFIBUS-DP и -PA применяются подмножества служб уровня 2. Так, например, PROFIBUS-DP использует только службы SRD и SDN.

Службы вызываются через точки доступа к службе SAP (Service Access Point) уровня 2 из вышестоящего уровня. В PROFIBUS-FMS используются эти точки доступа для адресации логических коммуникационных связей. В PROFIBUS-DP и -PA применяемые точки доступа строго

упорядочены. У всех активных и пассивных участников можно использовать параллельно несколько точек доступа.

Различаются точки доступа источника SSAP (Source Service Access Point) и точки доступа цели DSAP (Destination Service Access Point).

Прикладной уровень (Layer 7)

Уровень 7 модели ISO/OSI представляет в распоряжение пользователя полезные коммуникационные службы. Этот пользовательский уровень состоит в PROFIBUS из FMS (Fieldbus Message Specification) и LLI (Lower Layer Interface) слоев.

Профиль FMS. Коммуникационные службы FMS соответствуют функциям прибора, т.е. в FMS-профилях PNO определен необходимый объем функций для конкретных требований. Эти FMS-профили устанавливают, что приборы различных производителей имеют одни и те же коммуникационные функции.

Для FMS определены следующие профили:

Коммуникации между контроллерами (3.002). Этот коммуникационный профиль устанавливает, какие FMS-службы применяются для коммуникаций между PLC. При помощи точно определенных классов контроллеров установлены службы, параметры и типы данных, которые каждый PLC должен поддерживать.

Профиль для автоматизации зданий (3.011). Этот профиль – отраслевой (специализированный) профиль и основа для многих открытых стандартов в автоматизации зданий. Описывает, как осуществляется обмен, управление, регулирование, обслуживание, обработка и архивирование сигналов (Alarm) в системах автоматизации зданий через FMS.

Коммутационные низковольтные приборы (3.032). Этот профиль – отраслевой пользовательский FMS-профиль. Он определяет пользовательский образ действий низковольтных коммутационных приборов при коммуникациях через FMS.

Пользовательский интерфейс DP и DP-профили. PROFIBUS-DP применяет уровни 1 и 2. Через пользовательский интерфейс становятся доступными необходимые пользовательские функции, а также системные и аппаратные действия различных типов устройств PROFIBUS-DP.

Протокол PROFIBUS-DP открыто определяет, как передаются между участниками пользовательские данные по шине. Оценка передаваемых по

протоколу пользовательских данных не происходит. Благодаря точно установленным параметрам профиля приборы разных производителей могут совершать обмен.

В настоящее время установлены следующие профили PROFIBUS-DP:

Профиль для NC/RC (3.052). Профиль описывает, как происходит управление и обслуживание роботов через PROFIBUS-DP. На основании конкретной блок-схемы программы описывается движение и программное управление роботом.

Профиль для Encoder (преобразователь угол-код) (3.062). Профиль описывает присоединение различных Encoder к PROFIBUS-DP. Определены два аппаратных класса основных и дополнительных функций, таких как, например, масштабирование сигналов и расширенная диагностика.

Профиль для приводов с изменяемым числом оборотов (3.072). Ведущие производители техники приводов разработали общий PROFIDRIVE-профиль. Профиль устанавливает, как приводы параметрируются и передают заданные и истинные значения. Благодаря этому становится возможным обмен данными приводов различных производителей.

Профиль содержит необходимые установки для вида работы регуляторов числа оборотов и позиционирования. Профиль устанавливает основные функции приводов и дает достаточное свободное пространство для специфических пользовательских расширений. Профиль содержит описание пользовательских функций DP или альтернативных функций FMS.

Профиль для управления и наблюдения HMI (Human Machine Interface) (3.082). Профиль устанавливает для приборов обслуживания и наблюдения (HMI) правила подключения этих приборов через PROFIBUS-DP к компонентам автоматизации.

Профиль использует для коммуникаций расширенные функции PROFIBUS-DP.

Профиль для защищенной от ошибок передачи данных через PROFIBUS-DP (3.092). В этом профиле устанавливаются дополнительные механизмы защиты данных для коммуникаций с защищенными от ошибок компонентами, как, например, Not-AUS.

Топология шины

RS-485-техника

Технологически система PROFIBUS состоит из нагруженной с двух сторон активной линии – шинной структуры, которая обозначается так же, как сегмент шины RS-485. К шинному сегменту можно по стандарту RS-485 подключить до 32 участников. Каждый подключенный к шине участник, Master или Slave, представляет собой токовую нагрузку.

Повторитель (Repeater). Если Вы должны подключить к системе PROFIBUS больше чем 32 участника, то нужно использовать несколько шинных сегментов. Эти отдельные шинные сегменты, каждый максимум с 32 участниками, должны быть соединены друг с другом через повторитель (усилитель мощности). Повторитель усиливает уровень передаваемого сигнала. Согласно EN 50170 не предусмотрена временная регенерация фазы бита во время передачи сигнала через повторитель. Из-за временных задержек и искажений двоичный сигнал может согласно EN 50170 проходить максимум три повторителя, которые работают как усилители мощности и включены последовательно. На практике, однако, повторитель-соединитель реализуется как восстановитель сигнала. Число повторителей, которые можно включить последовательно, таким образом, зависит от конструкции и изготовителя. Так, например, можно последовательно включить до 9 повторителей типа 6ES7 фирмы Siemens.

Максимальное удаление между двумя участниками шины зависит от скорости передачи. В табл. 6.9 даны значения для повторителя типа 6ES7.

Таблица 6.9

Максимально возможное расширение конфигурации PROFIBUS
при включенных в ряд 9 повторителей
в зависимости от скорости передачи

Скорость передачи, кбит/с	9,6 – 187,5	500	1500	12000
Общая длина всех сегментов, м	10000	4000	2000	1000

Принципиальная схема, изображенная на рис. 6.20, поясняет свойства RS-485-повторителя. Шинный сегмент 1, гнездо PG/PC и шинный сегмент 2 отделены друг от друга потенциалами. Сигнал между шинным сегментом 1, гнездом PG/PC и шинным сегментом 2 усиливается. Повторитель имеет для шинных сегментов 1 и 2 подключаемое сопротивление. Благодаря разделителю – мосту М/РЕ – повторитель может работать без заземления.

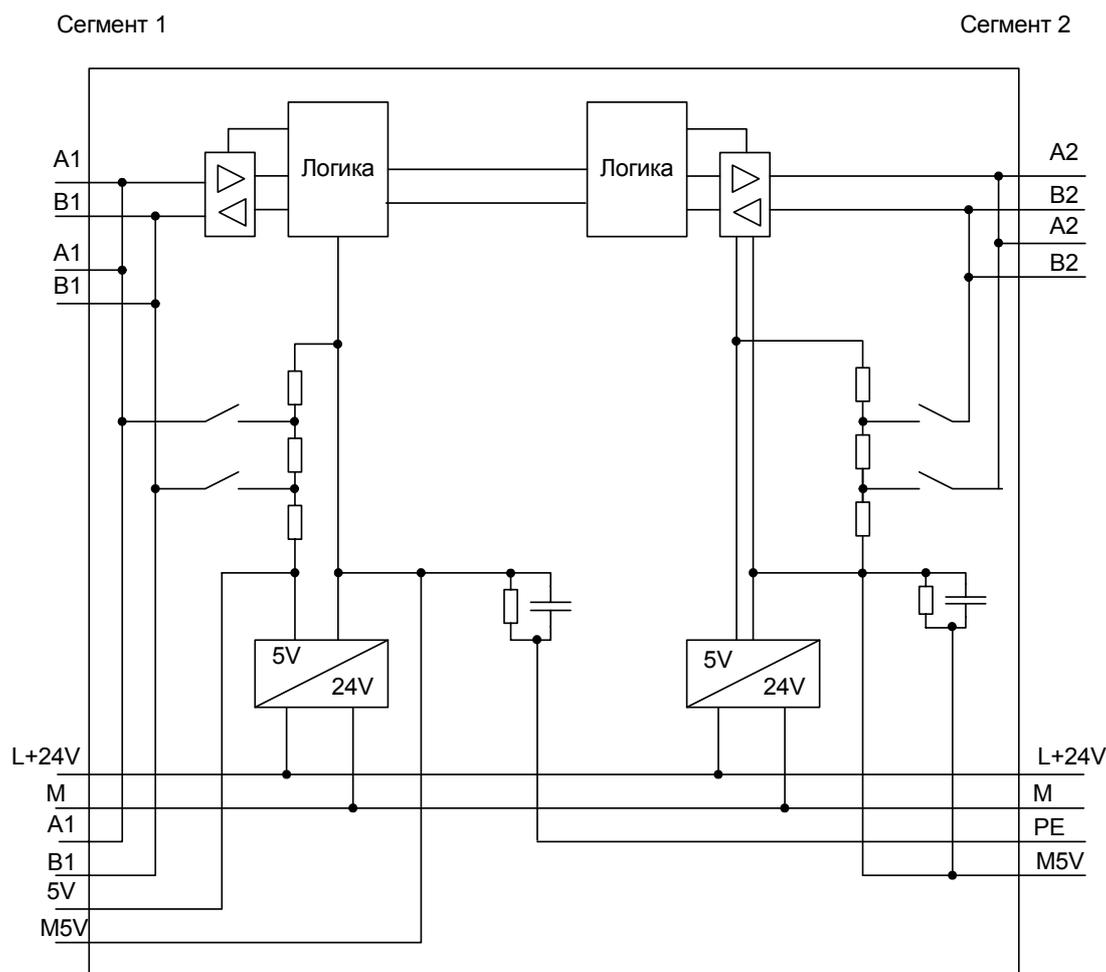


Рис. 6.20. Принципиальная схема RS-485-репитера типа 6ES7

Только благодаря использованию повторителя может быть достигнуто максимально возможное число участников в конфигурации PROFIBUS. Повторитель можно применять также для построения шинных структур типа “дерево” или ”звезда”.

Можно создать свободные от заземления структуры (разделение шинных сегментов друг от друга) с помощью повторителя и источника питания 24 В без заземления (рис. 6.21).

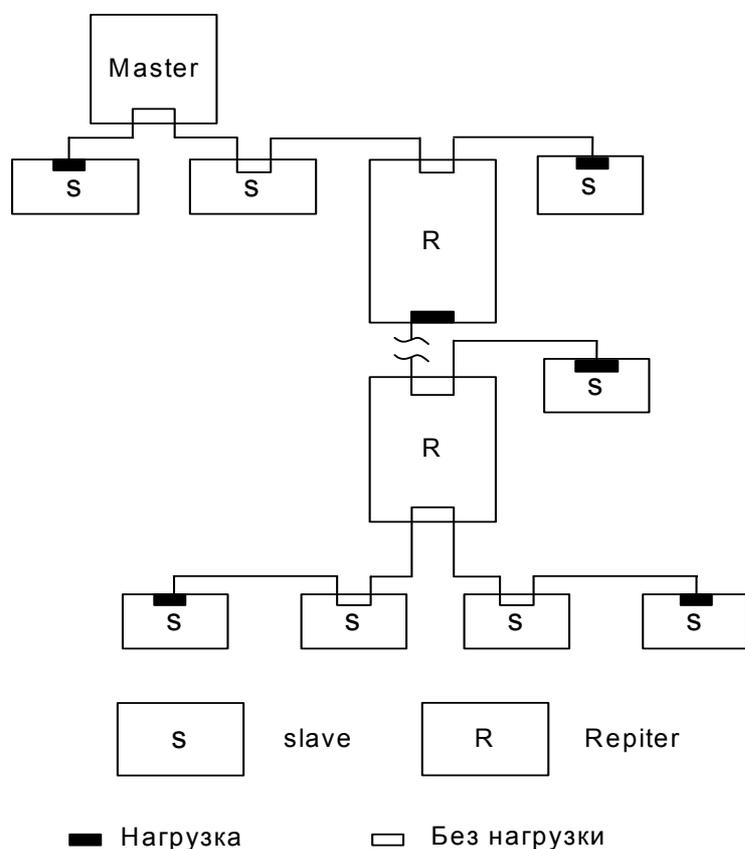


Рис. 6.21. Шинная конфигурация с повторителем

логического шинного адреса).

Четвертьволновые отрезки линии передачи. Благодаря прямому подключению участников, например, через 9-штырьковый штекер, в линейной структуре возникают четвертьволновые отрезки линии передачи. По EN50170 при скорости передачи 1500 кбит/с допустима длина отрезка не более 6,6 м. Как правило, эти отрезки должны быть еще уменьшены.

Исключение здесь представляет временное подключение устройств программирования и диагностики.

Повторитель также представляет нагрузку для соединения RS-485. Подключенный RS-485-повторитель уменьшает на 1 максимальное число участников на сегменте. Это значит, что если на шинном сегменте находится повторитель, то можно на этот сегмент подключить максимум 31 участника. Число повторителей в общей шинной конфигурации не влияет на максимальное число участников (повторитель не занимает

Четвертьволновые отрезки линии передачи могут в зависимости от числа и длины быть причиной переотражений и приводить к искажениям телеграмм.

При скорости передачи более 1500 кбит/с четвертьволновые отрезки линии передачи недопустимы. Программаторы и приборы диагностики могут быть в этом случае подключены через “активные” шинные соединители.

Оптоволоконная техника. При использовании оптоволоконной техники имеются уже готовые шинные структуры, такие как линия, дерево, звезда, различные варианты кольцевой структуры. С помощью OLM (Optic Link Module) можно реализовать как однопроводное (однофазное) кольцо, так и резервированное двухпроводное (двухфазное) кольцо (рис. 6.22).

При однопроводной структуре OLM соединяются друг с другом одинарным оптическим кабелем. Если появляется неисправность, как, например, разрыв оптоволокну или выход из строя OLM, кольцо разрывается. При резервированном оптоволоконном кольце OLM соединяются друг с другом двумя дуплексными оптическими кабелями. В этом случае система может сама распознать неисправность и сконфигурировать шинную систему в линейную структуру.

Сообщения о повреждениях поступают через соответствующие контакты и могут быть обработаны. Как только разрыв оптоволокну будет устранен, шинная система опять конфигурируется в резервированное кольцо.

Техника по IEC 1158-2 (PROFIBUS-PA). С помощью PROFIBUS-PA могут быть реализованы отдельные структуры: линейные, древовидные, звездообразные, а также их комбинации.

Количество шинных сегментов, занятых участниками шины, зависит от установленных источников питания, тока, потребляемого участниками, типа кабеля и экрана шинной системы. На шинную систему можно подключить до 32 участников. Чтобы повысить надежность системы, можно сделать сегмент резервированным. Подключение шинных сегментов PA к сегменту PROFIBUS-DP осуществляется с помощью сегментных соединителей (Segment-kopplers) (рис. 6.23) или DP/PA-Link.

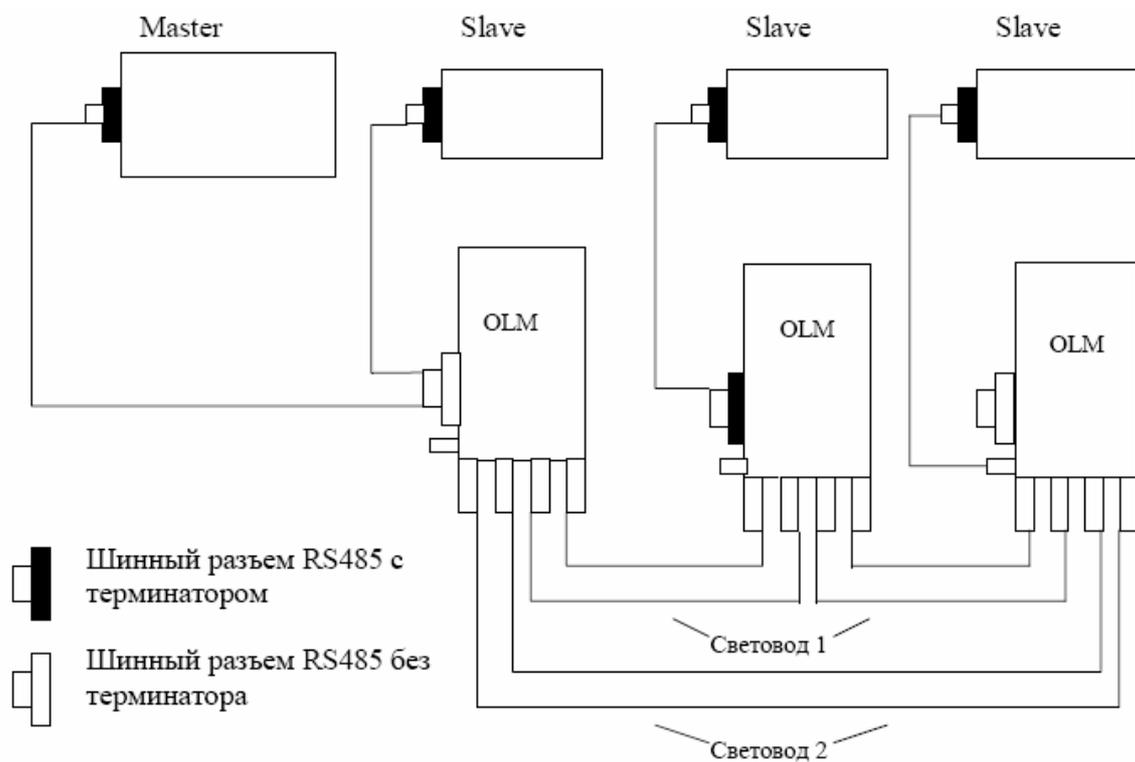


Рис 6.22. Резервированное двухпроводное (двухфазное) кольцо:

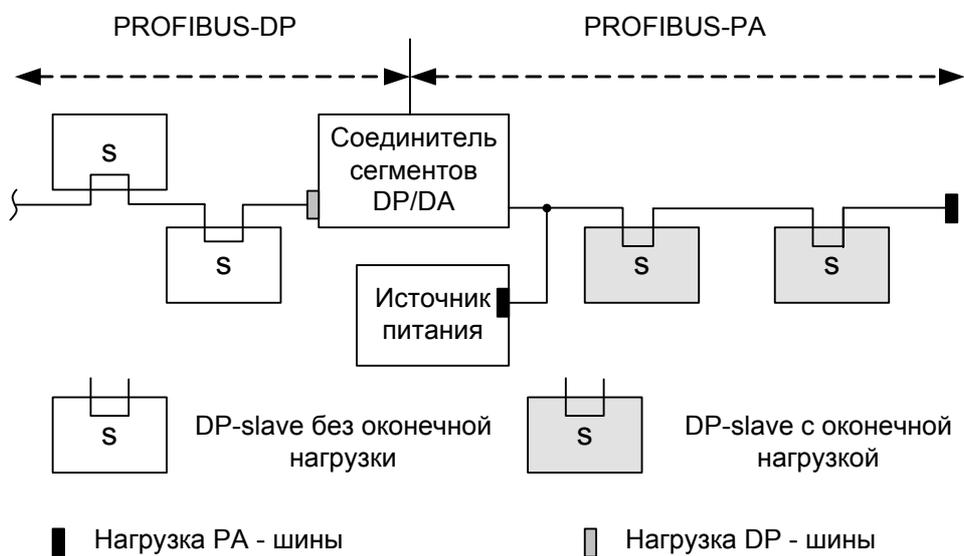


Рис. 6.23. Конфигурация шины с соединителем сегментов DP/PA:

Управление доступом к шине в PROFIBUS

К управлению доступом к шине PROFIBUS предъявляются два существенных требования.

С одной стороны, для надежных коммуникаций между равноправными приборами автоматизации или РС необходимо, чтобы каждый участник в течение определенного временного окна получал доступ к шине для решения своих коммуникационных задач.

С другой стороны, для обмена данными между сложными приборами автоматизации или РС и простой децентрализованной периферией требуется быстрый обмен данными с возможно малыми издержками протокола. Это достигается благодаря гибридно построенному управлению доступом к шине, состоящим из децентрализованного обмена маркером (токеном) между активными участниками (Master) и централизованного обмена Master - Slave для обмена данными между активными и пассивными участниками шины PROFIBUS.

Активный участник, который владеет маркером, берет на себя в данное время функции мастера на шине, чтобы проводить коммуникации с пассивными и активными участниками.

Обмен сообщениями по шине происходит при этом через адресацию участников. Каждому PROFIBUS-участнику назначается однозначный адрес. Адрес назначается из области от 0 до 126. При этом максимальное число участников, находящихся на шине, не превышает 127.

С этим управлением доступом к шине могут быть реализованы следующие конфигурации системы:

- “чистая” система Master – Master (обмен маркером);
- “чистая” система Master – Slave;
- Комбинация обоих методов.

Метод доступа к PROFIBUS не зависит от используемой среды передачи (например, медный провод или оптоволокно) и соответствует EN 50170.

В методе обмена маркером активные участники, подключенные к PROFIBUS, упорядочены по возрастанию их адреса в логическое маркерное кольцо (Token ring) (рис. 6.24).

Под маркерным кольцом (Token ring) здесь понимается организационное кольцо из активных участников, в котором маркер (Token) всегда передается от одного участника к следующему. Маркер, а с ним и право на доступ к среде передачи, передается при этом через специальную маркер-телеграмму между активными участниками. Исключение составляет активный участник с наивысшим на шине адресом HSA (Highest Station Ad-

dress). Он передает маркер исключительно активному участнику с наименьшим шинным адресом, чтобы замкнуть маркерное кольцо.

Время одного обращения маркера через всех активных участников называется *временем обращения маркера*. С помощью устанавливаемого заданного времени обращения маркера TTR (Time Target Rotation) определяется максимально разрешенное время обращения маркера.

Управление доступом к шине активных участников (MAC – Medium Access Control) осуществляется как на фазе инициализации, так и на фазе функционирования маркерного кольца. При этом устанавливаются адреса всех имеющихся на шине активных участников и заносятся в LAS (List of Active Station – список активных станций). Для управления маркером при этом особенно важны адреса предыдущей станции PS (Previous Station), от которой маркер получается, и следующей станции NS (Next Station), которой маркер предназначается. Кроме того, LAS также нужна, чтобы при текущей работе исключать из кольца вышедших из строя или дефектных активных участников и, соответственно, принимать вновь появившихся участников без помех текущему обмену данными по шине.

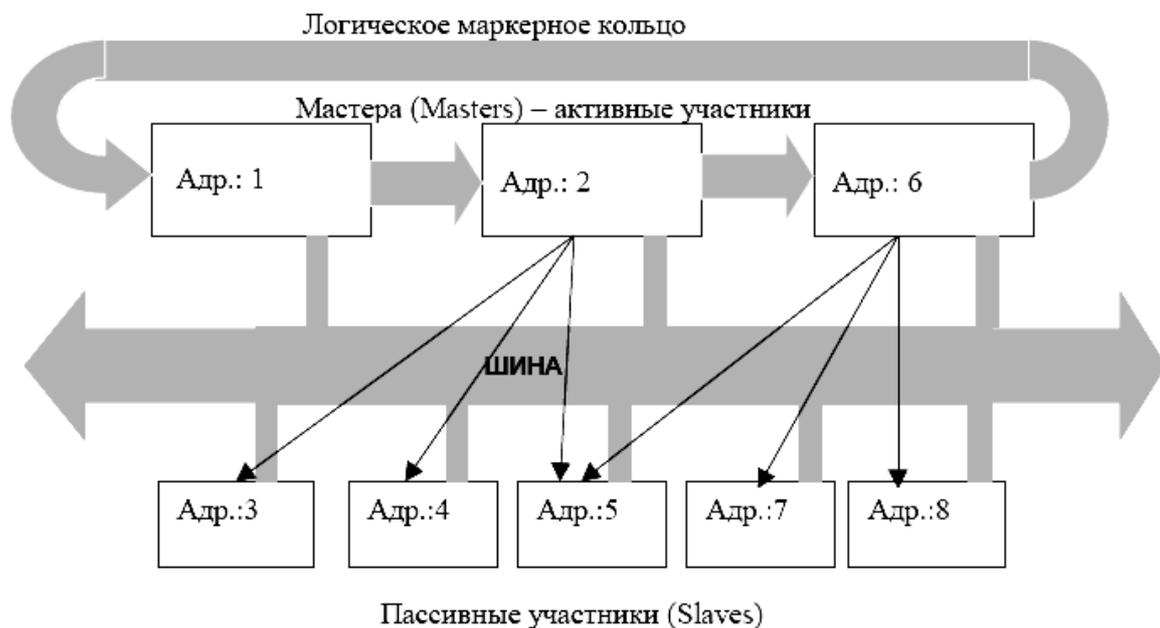


Рис.6.24. Метод обмена маркером (токеном)

Если логическое маркерное кольцо состоит только из одного активного и нескольких пассивных участников, то это соответствует “чистой” системе Master – Slave (рис. 6.24).

Метод Master – Slave дает возможность Master (активному участнику), который имеет право прямой передачи, опрашивать назначенных ему Slaves (пассивных участников). Master при этом имеет возможность принимать сообщения от Slave и соответственно передавать.

Типичная стандартная шинная конфигурация PROFIBUS-DP базируется на этом методе управления шиной. Активная станция (DP- Master) обменивается в циклической последовательности данными с пассивными станциями (DP-Slaves).

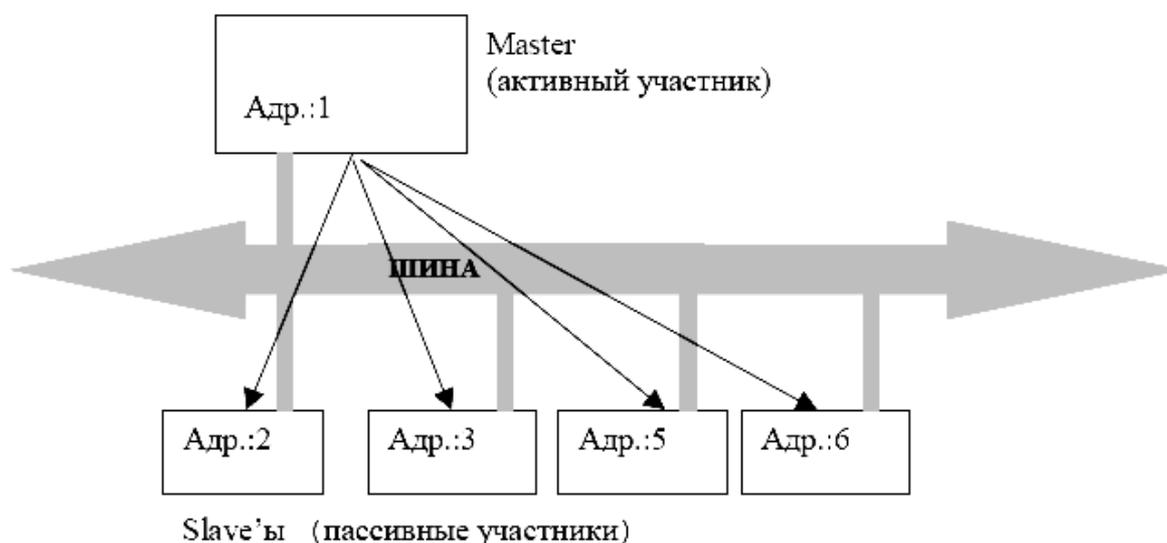


Рис. 6.25. Метод доступа Master – Slave

Контрольные вопросы

1. Как реализован физический уровень CAN-протокола и какими физическими характеристиками он обладает?
2. Является ли CAN-протокол общей средой передачи данных и что это означает?
3. Чем отличается CAN-протокол от CAN open и как реализуется передача данных?
4. Какие типы сообщений используются в CAN-протоколе?
5. Из каких элементов состоит сообщение Data Frame?
6. В чем заключается механизм работы побитового арбитража и способы обнаружения ошибок?

7. Назовите основные положения стандарта CAN.
8. Опишите типы фреймов, используемые в протоколе CAN.
9. Каким образом реализуется адресация в протоколе?
10. Какие виды ошибок определяет CAN-протокол?
11. Для каких целей используются прерывания и каков механизм их действия?
12. Какие возможности предоставляет протокол при реализации задач управления в реальном времени?
13. Как построен программный интерфейс, обеспечивающий доступ к среде?
14. Перечислите библиотечные функции программного интерфейса и дайте их краткую характеристику.
15. Что представляет собой протокол Device Net?
16. Какие типы идентификаторов, поддерживающих адресацию, используются в сети Device Net? Назовите их.
17. Опишите объектную модель сетевых ресурсов.
18. Перечислите объекты, входящие в объектную модель.
19. Какие уровни модели OSI/ISO используются в стандарте PROFIBUS?
20. Какой из стандартов последовательной передачи данных использует PROFIBUS?
21. Опишите структуру шинного сегмента PROFIBUS и укажите способ передачи информации.
22. Какими техническими характеристиками обладает шина PROFIBUS?
23. Какими средствами обеспечивается подключение к шине PROFIBUS?
24. Каким требованиям должен отвечать кабель, используемый для реализации шин?
25. Какие форматы кадров используются в протоколе PROFIBUS и как распределяется информация в пределах кадра?
26. Назовите профили PROFIBUS-DP, используемые для автоматизации технологических процессов.
27. Опишите процедуры работы повторителя для PROFIBUS.
28. Как обеспечивается согласование линии и обеспечивается надежность ее работы?
29. Назовите методы доступа к шине и дайте им краткую характеристику.

Глава 7. ЛОГИЧЕСКАЯ ОРГАНИЗАЦИЯ СЕТИ

Рассматривать логическую организацию сети в отрыве от реального объекта возможно, но для понимания работы сети необходимо опираться на конкретную реализацию. Если для физической реализации среды передачи данных процесс управления не представляет интереса, то для логической он является принципиальным. Поэтому возьмем в качестве примера схему, представленную на рис. 7.1.

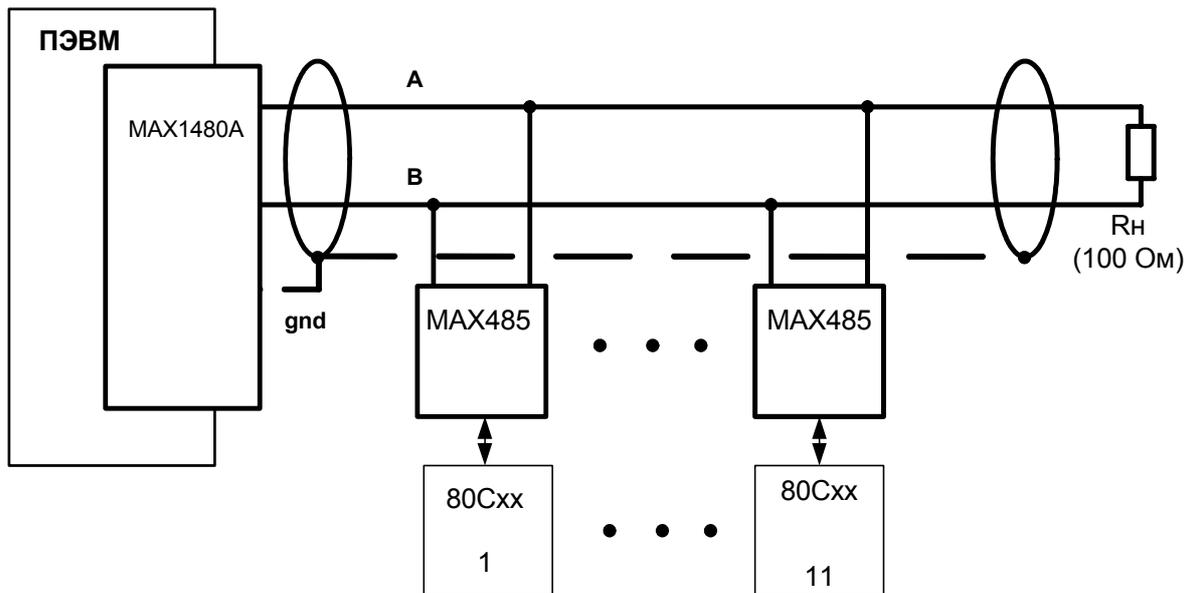


Рис. 7.1. Схема сети локальных контроллеров

Сеть построена на основе общей шины и включает в себя персональную ЭВМ и одиннадцать контроллеров (МК), построенных на основе микроЭВМ. Восемь из них управляют работой шаговых двигателей, а три оставшихся – приводами постоянного тока. Каждый из контроллеров выполняет стандартный набор функций. Так, для контроллера, управляющего работой шагового двигателя, необходимо передавать информацию о скорости, направлении вращения и величине перемещения. В ответ на выполняемую операцию необходимо получать информацию о величинах действительного перемещения. Кроме того, перед началом функционирования привода необходимо либо получить информацию о его начальном положении и готовности к работе, либо принудительно, по команде, установить его в начальное положение. Это аналогично и для приводов постоянного тока.

Приводы в данной системе могут работать в автоматическом, ручном и наладочном режимах.

Облегчение анализа работы связано с типовым решением шаговых приводов и приводов постоянного тока. Иными словами, для создания протокола достаточно рассмотреть работу одного из приводов на основе шагового двигателя и одного привода постоянного тока.

В предыдущих главах рассмотрена работа ЭВМ при использовании коммуникационного порта. Но поскольку все процессы в рассматриваемой системе инициируются ЭВМ, отметим ряд особенностей, связанных с анализом работы линии связи и использовании этой информации для реализации протокола. ПК связан с микроЭВМ посредством специализированной микросхемы MAX1480A, обеспечивающей переход из протокола RS-232C в протокол RS-485.

Если принять перепад напряжения в линии, сигнализирующий о начале посылки, а по стробу старт-бита зафиксировать уровень логической единицы, старт-бит считается ложным и приемник снова переходит в состояние ожидания. Об этой ошибке приемник может не сообщать. Если во время, отведенное под стоп-бит, обнаружен уровень логического нуля, фиксируется ошибка стоп-бита. Если применяется контроль четности, то после посылки бит данных передается контрольный бит. Этот бит дополняет количество единичных бит данных до четного или нечетного в зависимости от принятого соглашения. Прием байта с неверным значением контрольного бита приводит к фиксации ошибки.

Контроль формата позволяет обнаруживать обрыв линии: как правило, при обрыве приемник «видит» логический нуль, который сначала трактуется как старт-бит и нулевые биты данных, но потом срабатывает контроль стоп-бита. Для асинхронного режима принят ряд стандартных скоростей обмена 50, 75, 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600 и 115200 бит/с. Иногда вместо единицы измерения «бит/с» используют «бод» (baud), но при рассмотрении передаваемых двоичных сигналов это некорректно. В бодах принято измерять частоту изменения состояния линии, а при недвоичном способе кодирования (широко применяемом в современных модемах) в канале связи скорости передачи бит (бит/с) и изменения сигнала (бод) могут отличаться в несколько раз.

Количество бит данных может составлять 5, 6, 7 или 8 (5- и 6-битные форматы распространены незначительно). Количество стоп-бит может

быть 1; 1,5 или 2 (полтора бита означает только длительность стопового интервала).

Компьютер может иметь до четырех последовательных портов COM1 – COM4. Для машин класса AT типично наличие двух портов. Управление последовательным портом разделяется на два этапа – предварительное конфигурирование (Setup) аппаратных средств порта и текущее (оперативное) переключение режимов работы прикладным или системным ПО. Конфигурирование COM-порта зависит от его исполнения. Порт на плате расширения конфигурируется «джамперами» на самой плате. Порт на системной плате конфигурируется через BIOS. Конфигурированию подлежат перечисленные ниже параметры, базовый адрес, который для портов COM1 – COM4 обычно имеет значение 3F8h, 2F8h, 3E8h и 2E8h. При инициализации BIOS проверяет наличие портов по адресам именно в этом порядке и присваивает обнаруженным портам логические имена COM1, COM2, COM3 и COM4. Для портов COM3 и COM4 возможны альтернативные адреса 3E0h, 338h и 2E0h, 238h соответственно. Для PS/2 стандартными для портов COM3 – COM8 являются адреса 3220h, 3228h, 4220h, 4228h, 5220h и 5228h соответственно.

Для COM1 и COM3 обычно используется линия запроса прерывания IRQ4 или IRQ11, для COM2 и COM4 — IRQ3 или IRQ10. В принципе номер прерывания можно назначать в произвольных сочетаниях с базовым адресом (номером порта), но некоторые программы и драйверы (например, драйверы последовательной мыши) настроены на стандартные сочетания. Каждому порту, нуждающемуся в аппаратном прерывании, назначают отдельную линию, не совпадающую с линиями запроса прерываний других устройств. Прерывания необходимы для портов, к которым подключаются устройства ввода, UPS или модемы. При подключении принтера или плоттера прерываниями пользуются только многозадачные ОС (не всегда), и этот дефицитный ресурс ПК можно сэкономить. Также прерывания обычно не задействуют при связи двух компьютеров нуль-модемным кабелем. Возможность разделяемого использования одной линии запроса несколькими портами (или ее разделения с другими устройствами) зависит от реализации аппаратного подключения и ПО. При использовании портов, установленных на шину ISA, разделяемые прерывания обычно не работают.

В рассматриваемой системе управления организован принцип доступа к среде с ведущей ЭВМ. Все микроЭВМ являются ведомыми устройст-

вами в сети. Между собой микроЭВМ общаться не могут. Весь обмен информацией осуществляется через ведущую ЭВМ (персональный компьютер). Микроконтроллеры могут отвечать только на запрос от ведущей ЭВМ. Когда ПК не активен, на линии нет сигналов.

Модуль управления может быть написан на языке Visual Basic NET, используя стандартные средства доступа к последовательному порту, функции WIN32API, которые, в свою очередь, обращаются к драйверу WINDOWS, работающему напрямую с портами ввода – вывода. Модуль (файл Rs232.vb) представляет собой полноценный пользовательский объектный класс, имеющий свои свойства, методы и события. Свойства представляют собой обычные локальные переменные, которые хранят настройки COM-порта, такие как скорость передачи (в нашей среде скорость – 56700 бит/с), количество стоп-битов (равно 0), контроль чётности (не включен) и многое другое. Методы данного класса включают в себя процедуры для работы с портом посредством WIN32API-функций, таких как инициализация порта, чтение состояния порта, запись и чтение (синхронное и асинхронное), чтение ошибок на линии, чтение текущих настроек и т.д. События представляют собой прерывания, процедуры, возникающие на чтение первого байта, на ошибку в посылке данных, на превышение тайм-аута. Этот класс полностью локальный и не нуждается в каких-либо глобальных переменных, т.е. его можно вставить в любой проект как объект. Так как это полноценный объект, то он обладает свойствами наследования и размножения, а следовательно, теоретически модуль может одновременно поддерживать несколько открытых портов и соответственно несколько сетей. Практически в системе управления реализовано только однопортовое управление. Вообще говоря, передаваемые данные можно оформлять различными способами. Формат передаваемых данных можно представлять как кадр постоянной длины, переменной длины и постоянной длины с блоком данных. Структура кадра может включать в себя байт-идентификатор команды, байт адреса источника команды, выполняемой микропроцессором, байт спецификации, контрольную сумму и байт конца кадра (рис. 7.2).

Механизм взаимодействия микроЭВМ с ПК основан на потоковой передаче информации. Поток состоит из 14 байт. Число байт определяется особенностью работы контроллеров и самой системы управления.

Регламент работы сети определяется протоколом, и зависит от типа команды выдаваемой ПК.

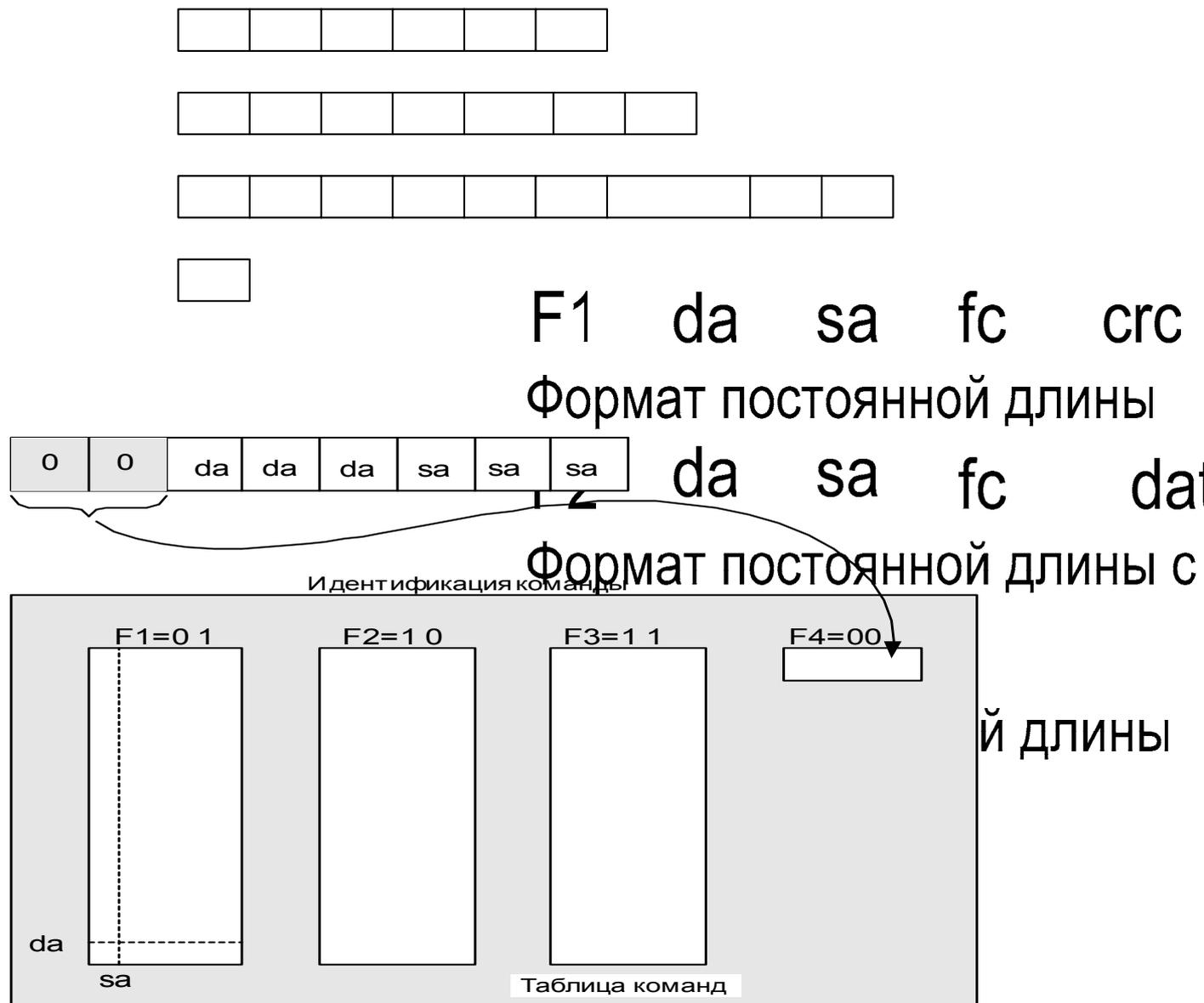


Рис. 7.2. Возможные структуры кадров:

da – приемник данных; *sa* – источник данных; *fc* – байт спецификации, *crc* – контрольная сумма; *data* – данные; *end* – конец передачи; *F_x* – код команды

Особенности построения сети по схеме общей шины, накладывают ряд неудобств на работу всей СУ. Отметим, что в результате передачи данных от ПК в сеть, из-за того, что последовательный порт МК работает в режиме прерываний, с приходом байта по последовательной шине, МК на

время чтения 14 байт данных теряет управление над текущим состоянием привода, подключенного к нему через порты ввода-вывода. Это негативно сказывается на стабильности вращения двигателей или контроле траектории СУ. Обязательный приём всех 14 байт обусловлен синхронизацией всех контроллеров, по 1 стартовому байту в пакете (номер контроллера). Для отделения (фильтрации) контрольной суммы от адреса микроЭВМ старший бит CRC устанавливается в «1» принудительно.

7.1. Протокол работы сети

Структура пакета (кадра) (табл. 7.1) включает в себя адрес устройства в сети, режим работы управляемого устройства, специальные команды для контроллера в сети, набор служебной информации и обязательный байт контрольной суммы. Расположение байт в пакете строго определенное и не может изменяться. Не может изменяться и число байт, передаваемых в сообщении.

Таблица 7.1

Структура пакета

Порядковый номер байта в пакете	Назначение	Размерность, байт
1	АДРЕС – адрес контроллера на шине	1
2	РЕЖИМ – режим работы	1
3	КОМАНДА – команды для микроЭВМ	1
4...13	Служебная информация	9
14	CRC – контрольная сумма пакета	1

Рассмотрим работу шаговых приводов. Привод обеспечивает: вращение исполнительных органов влево и вправо; перемещение на определенную величину, задаваемую пропорционально углу поворота вала двигателя; скорость вращения. Привод должен работать в ручном, автоматическом и наладочном режимах. Кроме того, привод имеет концевые выключатели, определяющие конечные положения слева и справа и начальное положение, от которого ведется отсчет.

Процедура обмена информацией при реализации протокола предполагает передачу информации в двух направлениях. Это показано на

рис. 7.3, а. От ПК передается информация о скорости вращения и величине перемещения, а от МК возвращается информация о выполнении заданных функций. В том и другом случае выполняется квитирование, что позволяет гарантировать достоверность циркулирующей в сети информации. Особенности работы протокола сведены в табл. 7.2.

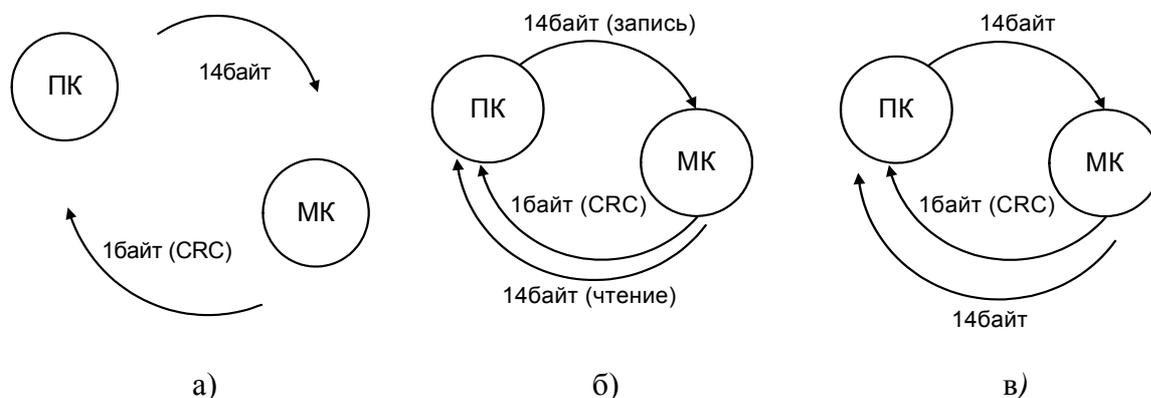


Рис. 7.3. Передача информации при реализации протокола управления:
 а – для шаговых двигателей; б – для привода постоянного тока;
 в – для релейной автоматики

Таблица 7.2

Пакет управления шаговыми двигателями

Но- мер байта	Имя	Описание	Допустимые значения, константы
1	$N_{\text{прив}}$	Адрес шагового при- вода на шине	1 – 8
2	Rejim	Режим работы	5 – автомат 4 – ручной 3 – полуавтомат
3	Command	Команда	8 – движение в положи- тельном направлении (по часовой стрелке) 9 – движение в отрица- тельном направлении (про- тив часовой стрелки) 10 – выход в начальную точку 12 – стоп

Но- мер байта	Имя	Описание	Допустимые значения, константы
4	V _{drive}	Скорость двигателя	0 – 255
5	0	Не используется	
6	AngleML	Угол поворота (млад- ший байт в автомати- ческом режиме)	0 – 255
7	AngleST	Угол поворота стар- ший байт (в автомати- ческом режиме)	0 – 255
8	0	Не используется	-
9	0	»»»	-
10	0	»»»	-
11	0	»»»	-
12	0	»»»	-
13	0	»»»	-
14	CRC	Младший байт кон- трольной суммы со 2- го по 13-й байты вклю- чительно. Старший бит принудительно уста- навливается в «1»	128 – 255

Команда считается выполненной успешно, если один принятый байт от микроЭВМ совпадает с контрольной суммой отосланного пакета.

7.2. Управление дискретными сигналами

В системе управления используются устройства для управления которыми достаточно одного бита. Часть устройств выдают информацию о состоянии управляемого устройства в виде дискретных сигналов, которые также кодируются одним битом. Обработка этих сигналов ведется в отдельном контроллере, а затем передается в ПК. Эта информация то же яв-

ляется содержанием протокола обмена информацией. Назначение отправляемых и получаемых байт пакета приведено в табл. 7.3 и 7.4 соответственно. Количество байт, принимаемых от МК, зависит от типа команды.

Таблица 7.3

Пакет данных передаваемых микропроцессорной плате

Номер байта	Имя	Описание	Допустимые значения, константы
1	$N_{\text{прив}}$	Адрес микроконтроллера на последовательной шине	11
2	Rejim	Режим работы	5 – автомат 4 – ручной 3 – полуавтомат
3	Command	Команда	8 – запись значения состояния ключей (4 байта) в регистр данных 6 – прочитать данные
4	Vdrive	Состояние ключей	0 – 16
5	Selector	Адрес подплаты	234 – 235 – запись данных в регистр 198 – 199 – чтение данных
6	0	Не используется	-
7	0	»»»	-
8	0	»»»	-
9	0	»»»	-
10	0	»»»	-
11	0	»»»	-
12	0	»»»	-
13	0	»»»	-
14	CRC	Младший байт контрольной суммы со 2-го по 13-й байты включительно	128 – 255

Таблица 7.4

Пакет принимаемых данных от микропроцессорной платы

Но- мер байта	Имя	Описание	Допустимые значения, константы
1	$N_{\text{прив}}$	Адрес ПК на последовательной шине	12
2	Rejim	Режим работы	5 – автомат 4 – ручной 3 – полуавтомат
3	Command	Команда	6 – прочитайте данные.
4	Vdrive	Состояние ключей	0 – 16
5	Selector	Адрес подплаты	234 – 235 – запись данных в регистр 198 – 199 – чтение данных
6	0	Не используется	-
7	0	»»»	-
8	0	»»»	-
9	0	»»»	-
10	0	»»»	-
11	0	»»»	-
12	0	»»»	-
13	0	»»»	-
14	CRC	Младший байт контрольной суммы со 2-го по 13-й байт включительно	128 – 255

Последовательность чтения данных:

- ПК посылает пакет с командой «6»;
- МК принимает пакет, подсчитывает контрольную сумму и выставляет её на шину;
- МК читает с указанной подплаты данные и отправляет пакет для ПК.

Команда считается выполненной успешно, если контрольная сумма байт принятого пакета совпадает с последним байтом.

7.3. Управление двигателями постоянного тока, чтение датчиков положения

Привод постоянного тока, так же как и шаговый привод, выполняет функции перемещения на заданную величину и в двух заданных направлениях. Отличие заключается в том, что этот привод может работать как в позиционном режиме, так и в контурном. Передача команд для привода, работающего в контурном режиме, и данных выполняется без остановки привода, в то время как в позиционном режиме отработка перемещения осуществляется с остановкой привода. При управлении приводом в ручном режиме задается только скорость. Привод перемещается до тех пор, пока не будет задана команда на остановку. Во всех режимах работы необходимо учитывать аварийные ситуации, когда привод достигает крайних положений, контролируемых концевыми выключателями.

Направление вращения ДПТ определяется значением скорости (рис. 7.4): от 0 до 7FFh – положительное направление; от FFFh до 800h – отрицательное направление.

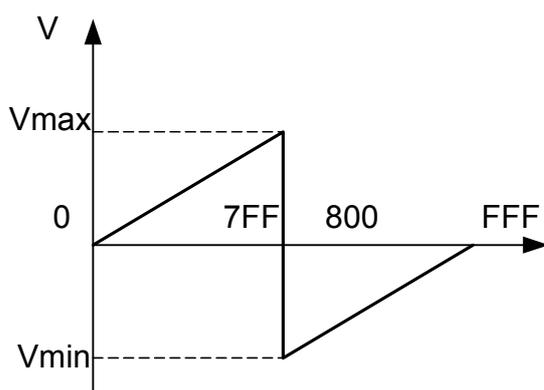


Рис. 7.4. Кодирование скорости

Управление ДПТ можно разделить на две большие группы: это ручной режим и автоматический режим (перемещение по контуру).

Протокол ручного режима включает в себя:

- запись в ЦАП код скорости двигателя по схеме на рис. 7.3, б (значения байтов передаваемого пакета представлено в табл. 7.5);
- чтение датчиков положения по схеме на рис. 7.3, в (значения принимаемых байт пакета представлено в табл. 7.6).

Таблица 7.5

Пакет передаваемых данных плате управления ДПТ

Но- мер байта	Имя	Описание	Допустимые значения, константы
1	$N_{\text{прив}}$	Адрес микроконтроллера на последовательной шине	10 – приводы осей X и Y 9 – привод оси Z
2	Rejim	Режим работы	4 – ручной

Но- мер байта	Имя	Описание	Допустимые значения, кон- станты
3	Command	Команда	13 – включить привод X, Z в «+» 14 – включить привод X, Z в «-» 3 – включить привод Y в «+» 4 – включить привод Y в «-» 15 – остановить привод X, Z 7 – остановить привод Y 6 – прочитать данные 10 – позиционировать привод X, Z в нулевую точку 5 – позиционировать привод Y в нулевую точку 1 – обнуление датчика по X, Z 2 – обнуление датчика по Y
4	VdriveML	Значение скорости (младший байт)	0 – 255
5	VdriveST	Значение скорости (старший байт)	0 – 255
6	0	Не используется	-
7	0	»»»	-
8	0	»»»	-
9	0	»»»	-
10	0	»»»	-
11	0	»»»	-
12	0	»»»	-
13	0	»»»	-
14	CRC	Младший байт контрольной суммы со 2-го по 13-й байты включительно	128 – 255

Команда считается выполненной успешно, если принятый байт (CRC) равен контрольной сумме отправленного пакета.

Таблица 7.6

Пакет чтения датчиков положения

Но- мер байта	Имя	Описание	Допустимые значения, кон- станты
1	$N_{\text{прив}}$	Адрес ПК на последовательной шине	12
2	Rejim	Режим работы	4 – ручной
3	Command	Команда	6 – прочитать данные
4	CT_XML	Значение счётчика координаты X, Z (младший байт)	0 – 255
5	CT_XST	Значение счётчика координаты X, Z (старший байт)	0 – 255
6	CT_YML	Значение счётчика координаты Y, (младший байт)	0 – 255
7	CT_YST	Значение счётчика координаты Y, (старший байт)	0 – 255
8	SOST_SQ	Состояние концевиков	0 – 255
9	0	Не используется	-
10	0	»»»	-
11	0	»»»	-
12	0	»»»	-
13	0	»»»	-
14	CRC	Младший байт контрольной суммы со 2-го по 13-й байты включительно	128 – 255

Последовательность чтения данных:

- ПК посылает пакет с командой «6»;
- МК принимает пакет, подсчитывает контрольную сумму и выставляет её на шину;
- МК читает данные с указанной подплаты и отсылает пакет для ПК.

Команда считается выполненной успешно, если контрольная сумма байт принятого пакета совпадает с последним байтом.

Протокол автоматического режима включает в себя информацию, приведенную в табл. 7.7 и 7.8.

Таблица 7.7

Пакет передачи данных плате управления ДПТ

Но- мер байта	Имя	Описание	Допустимые зна- чения, константы
1	N _{прив}	Адрес микроконтроллера на последовательной шине	10 – приводы осей X и Y 9 – привод оси Z
2	Rejim	Режим работы	5 – автоматический
3	Com- mand	Команда	11 – перемещение по контуру
4	VX_ML, VZ_ML	Значение скорости привода X, Z (младший байт)	0 – 255
5	VX_ST, VZ_ST	Значение скорости привода X, Z (старший байт)	0 – 255
4	VY_ML	Значение скорости привода Y (младший байт)	0 – 255
5	VY_ST	Значение скорости привода Y (старший байт)	0 – 255
8	CT_XML, CT_ZML	Координаты конечной точки по оси X, Z (младший байт)	0 – 255
9	CT_XST, CT_ZST	Координаты конечной точки по оси X, Z (старший байт)	0 – 255
10	CT_YML	Координаты конечной точки по оси Y (младший байт)	0 – 255
11	CT_YST	Координаты конечной точки по оси Y (старший байт)	0 – 255
12	CONF	Байт конфигурации	0 – 255
13	0	Не используется	–
14	CRC	Младший байт контрольной суммы со 2-го по 13-й байты включительно	128 – 255

Таблица 7.8

Пакет чтения с платы управления ДПТ в автоматическом режиме

Номер байта	Имя	Описание	Допустимые значения, константы
1	$N_{\text{прив}}$	Адрес ПК на последовательной шине	12
2	Rejim	Режим работы	5 – автоматический
3	Command	Команда	6 – прочитать данные
4	CT_XML	Значение счётчика координат X, Z (младший байт)	0 – 255
5	CT_XST	Значение счётчика координат X, Z (старший байт)	0 – 255
6	CT_YML	Значение счётчика координат Y, (младший байт)	0 – 255
7	CT_YST	Значение счётчика координаты Y, (старший байт)	0 – 255
8	0	Не используется	-
9	0	»»»	-
10	0	»»»	-
11	0	»»»	-
12	CONF	Байт конфигурации	0 – 255
13	SOST_SQ	Состояние концевиков	0 – 255
14	CRC	Младший байт контрольной суммы со 2-го по 13-й байты включительно	128 – 255

7.4. Программа управления протоколом

Модуль управления протоколом реализован в одном файле «Пульт.vb». Для записи и чтения данных в программе применяется асинхронный режим, использующий многозадачность ОС Windows, т.е. данные процедуры запускаются параллельным потоком и выполняются параллельно с основной программой.

Данный модуль отвечает за правильность работы сети согласно установленному протоколу. Все операции по управлению и обмену данными с МК проходят через функции и процедуры этого модуля. В этом модуле происходит инициализация класса управления COM-портом и начальная его установка - процедура **COMP_PORT_INIT()**. Для упрощения работы с данным модулем написана процедура **SET_DRIVER({список параметров})**, которая, в зависимости от параметров, запускает нужную процедуру из этого модуля.

В список параметров входит:

- **Rejim_Pult** – режим работы;
- **NumberDrive** – номер микроЭВМ;
- **Commanda** – команда;
- **rezerv()** – массив служебной информации (в зависимости от команды содержимое массива разное).

Почти все процедуры данного модуля запускают метод **AsyncWrite** из класса RS232, который отвечает за запись буфера данных в COM-порт. Для чтения данных из буфера приемника – передатчика используется метод **AsyncRead** того же класса.

В программе предусмотрена также обработка ошибок, возникающих по разным причинам (большой уровень ВЧ-помех, скачки напряжения и т.д.), реализованная посредством квитирования контрольной суммы посланного кадра. Чтение контрольной суммы (CRC) посланного пакета процедура **GetCRCfromLINE()**. Но так как ПК быстрее обрабатывает команды, то для синхронизации введена задержка в данной функции на 5 мс. Если после неё CRC-байт не был получен, то генерируется ошибка и выводится в окне ошибок. Это также относится и к команде чтения датчиков.



Рис. 7.5. Чтение 1 байта

Кроме ответного CRC ещё посылается и пакет с данными от датчиков. Отличие лишь в том, что задержка на чтение 15 байт данных равна 10 мс. Временная диаграмма процесса записи и чтения показана на рис. 7.5.

Контрольные вопросы

1. Чем отличается логическая организация сети от физической?
2. Поясните, каким образом можно анализировать ложное срабатывание в передаваемой посылке?
3. Как организована система управления с точки зрения взаимодействия вычислительных устройств?
4. Чем определяется объем передаваемой информации?
5. От чего зависит структура кадра?
6. Каким образом распределяется информация в пределах кадра?
7. Что означает байт квитирования?
8. Приведите пример распределения информации в передаваемом пакете.
9. Каким образом обеспечивается двухсторонняя передача информации?
10. Является ли фиксированный пакет оптимальным с точки зрения быстродействия системы управления? Поясните почему.

ЗАКЛЮЧЕНИЕ

Рассмотренные в книге системы связей не отражают всего многообразия методов и средств передачи информации. Развитие электронных технологий позволяет сегодня использовать иную организацию сетей. Например, системы инфракрасной связи и радиосвязи. Они не требуют физических каналов, а используют для целей передачи информации воздушную среду. Конечно, это порождает свои проблемы. Так, для инфракрасного способа передачи требуется прямая видимость передатчика и приемника, для радиосвязи – использование достаточно дорогостоящего оборудования и борьба с помехами. Еще один аспект, характерный для этих видов связи, – конфиденциальность информации. При реализации беспроводных систем связи большая задача отводится защите информации от несанкционированного доступа.

Эти виды связи достаточно редко используются в машиностроении и являются предметом самостоятельного рассмотрения. Однако в ряде случаев при использовании подвижных объектов, например автономных транспортных систем, такие средства связи оправданы.

Указанные в пособии виды физических каналов связи не налагают ограничений на общие методы и средства передачи данных и построения сетей. Они лишь дополняют (расширяют) список средств передачи данных.

Сведения о них имеются в литературе, и читатель без труда может получить исчерпывающую информацию по всем интересующим его вопросам.

Таким образом, при выборе системы связей надо исходить из тех условий, которые обеспечат необходимые характеристики, заложенные в проекте.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Локальные вычислительные сети. В 2 кн. Кн. 1. Принципы построения, архитектура, коммуникационные средства / под ред. С. В. Назарова. – М. : Радио и связь, 1994. – 206 с.
2. **Мячев, А. А.** Интерфейсы систем обработки данных : справочник / А. А. Мячев ; под ред. А. А. Мячева. – М. : Радио и связь, 1989. – 416 с. – ISBN 5-256-00315-5.
3. **Николайчук, О.** Интерфейс персонального компьютера и локальной сети микроконтроллеров / О. Николайчук // Приборы и техника эксперимента. – 2000. – № 3. – С. 38 – 40.
4. Печатающие устройства для персональных ЭВМ : справочник / Е. П. Бененсон [и др.] ; под ред. И. М. Витенберга. – М. : Радио и связь, 1992. – 208 с.
5. **Гук, М.** Аппаратные средства IBM PC : энциклопедия / М. Гук. – СПб. : Питер, 1996. – 212 с. – ISBN 5-94723-180-8.
6. **Браун, Р.** Справочник по прерываниям IBM PC : в 2 т. : пер. с англ. / Р. Браун, Дж. Кайл. – М. : Мир, 1994. – 558 с.
7. **Данкан, Р.** Профессиональная работа в MS-DOS : пер. с англ. / Р. Данкан. – М. : Мир, 1993. – 509 с.
8. **Пильщиков, В. Н.** Программирование на языке ассемблера IBM PC / В. Н. Пильщиков. – М. : Диалог-МИФИ, 1997. – 288 С.
9. **RS-422 and RS-485 [Электронный ресурс] : Application Note.** – B&B Electronics Mfg. Co. Inc (Rev. 1997). – Режим доступа : <http://www.bb-elec.com/bb-elec/literature/tech/485appnote.pdf>. – Загл. с экрана. – Яз. англ.
10. Правильная разводка сетей RS-485 [Электронный ресурс]. – Maxim's Application Note 763 (январь, 2001). – Режим доступа : <http://www.gaw.ru/html.cgi/txt/interface/rs485/app.htm>. – Загл. с экрана. – Яз. англ.
11. Интерфейсы последовательной передачи данных. Стандарты RS-422/RS-485 [Электронный ресурс]. – Режим доступа : http://www.cta.ru/pdf/1997-3/note1_1997_3.pdf. – Загл. с экрана. – Яз. рус.
12. Описание протокола MODBUS [Электронный ресурс]. – Режим доступа : http://www.idom.ru/files/Schneider/Info/Networks/MODBUS/Modbus_Rus.Doc. – Загл. с экрана. – Яз. рус.
13. Selecting and Using RS-232, RS-422, and RS-485 Serial Data Standards [Электронный ресурс] : Maxim's Application Note 723. – Режим доступа : http://www.maxim-ic.com/appnotes.cfm/appnote_number/723. – Загл. с экрана. – Яз. англ.
14. Explanation of Maxim RS-485 Features [Электронный ресурс]. – Maxim's Application Note 367. – Режим доступа : http://www.maxim-ic.com/appnotes.cfm/appnote_number/367. – Загл. с экрана. – Яз. англ.

Учебное издание

ВЕСЕЛОВ Олег Вениаминович

ЛОКАЛЬНЫЕ СЕТИ
МАЛЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

Учебное пособие

Редактор И.А. Арефьева
Технический редактор Н.В. Тупицына
Корректор Е.В. Афанасьева
Компьютерная верстка С.В. Павлухиной

ЛР № 020275. Подписано в печать 28.12.05.

Формат 60x84/16. Бумага для множит. техники. Гарнитура Таймс.
Печать на ризографе. Усл. печ. л. 12,32. Уч.-изд. л. 13,53. Тираж 100 экз.

Заказ

Издательство

Владимирского государственного университета.
600000, Владимир, ул. Горького, 87.