

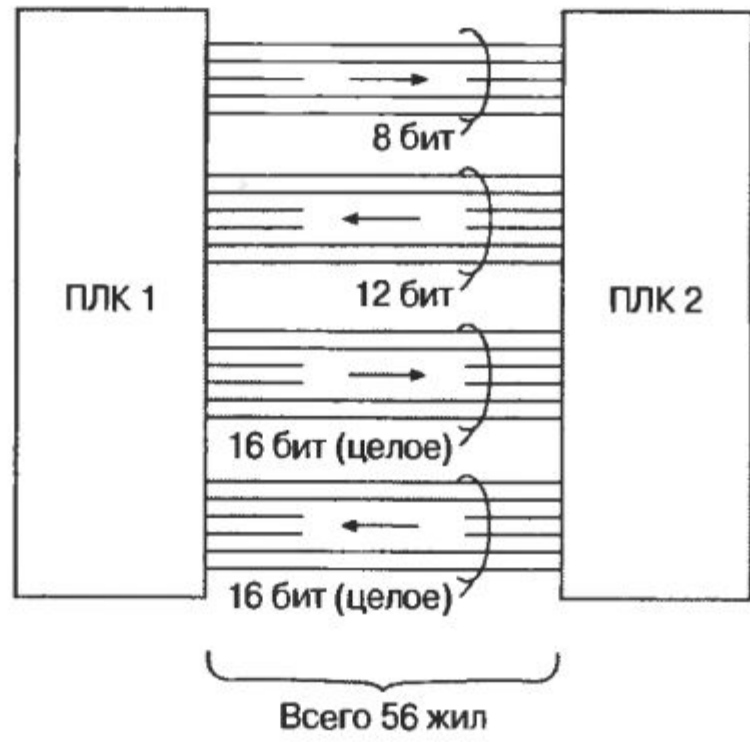
**МПСвЭПиТК**

Передача данных

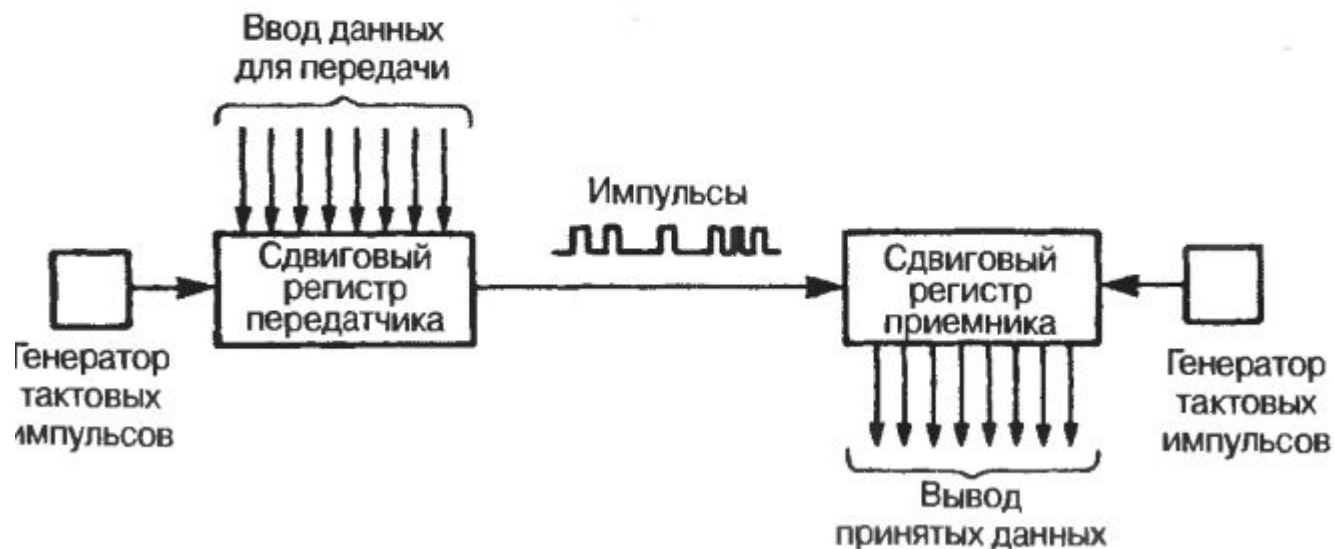
## **Параллельная и последовательная передача данных**

Монтаж проводки в любой системе управления требует наибольших затрат. Сюда относятся стоимость собственно кабелей, крепежных конструкций и кабельных желобов плюс затраты труда на прокладку кабелей и установку на концах разъемных соединений. Если в процессе сдачи системы в эксплуатацию обнаруживается, что необходимы линии для дополнительных сигналов, а резервные жилы отсутствуют, то потребуются прокладка дополнительного кабеля с вытекающими отсюда последствиями (лишние материальные затраты и задержка во времени).

На рис. 5.1 изображены два ПЛК, между которыми происходит обмен данными. Здесь 8 сигналов передаются в одном направлении и 12 в другом (все в виде напряжения 110 В переменного тока), а также два 16-битовых числа (в виде напряжения 24 В постоянного тока). С учетом шин питания, нейтральной шины и обратного провода по постоянному току всего для связи между ПЛК требуется 56 линий – возможно, один 27-жильный и один 37-жильный кабели, заключенные в металорукав, три 8-битовые цифровые выходные платы и столько же аналогичных входных плат (для обмена сигналами 110 В), две цифровые выходные платы и столько же аналогичных входных плат (для обмена сигналами 24 В). Необходимо затратить немало усилий для установки этих плат в корпусах на каждой стороне. Короче говоря, это не самый дешевый способ.



На рис. 5.2 единственная линия данных (не считая обратного провода) связывает передатчик и приемник, и по ней информация передается в виде последовательной строки битов. Поскольку компьютеры, периферийные устройства, ПЛК и т. д. работают с параллельным кодом (что повышает скорость обработки информации), то на передающей стороне необходимо осуществлять преобразова-



**Рис. 5.2.** Последовательная передача данных

ние параллельного кода в последовательный, а на приемной стороне — наоборот.

## **Стандарты последовательной передачи данных**

Для нормального функционирования системы последовательной передачи данных необходимо обеспечить согласованную работу передатчика и приемника. Должно быть четко указано следующее:

1. Уровни сигналов (по напряжению).
2. Код передачи (определение того, что означают битовые комбинации и как формируется сообщение).
3. Скорость передачи (скорость, с которой посылается битовая комбинация).
4. Синхронизация. На рис. 5.2 на каждом конце линии связи показаны генераторы тактовых импульсов. Если частота их работы отличается даже незначительно (а такие отличия неизбежны), то приемник не будет согласован с передатчиком.

Необходимо обеспечить тот или иной способ синхронизации работы передатчика и приемника.

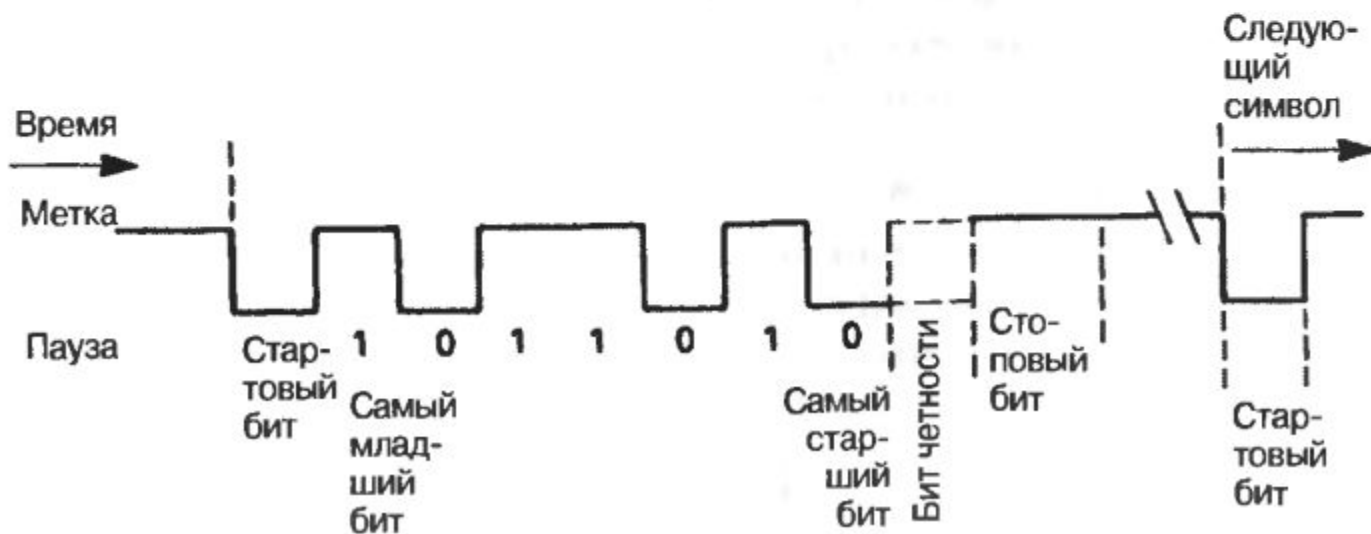
5. Протоколы. Не говоря уже о передаваемых данных, должен существовать некоторый метод обмена управляющими сигналами между передатчиком и приемником, типа «в данный момент я не могу принять сообщение».
6. Методы проверки на наличие ошибок и процедуры восстановления («последнее сообщение не имеет смысла, пожалуйста, передайте его еще раз»).

## Синхронизация

Самый простой метод синхронизации состоит в том, чтобы иметь один общий генератор тактовых импульсов для передатчика и приемника, т. к. два, даже теоретически, никогда не могут быть согласованы. Этот метод известен под названием синхронной передачи.

Большинству систем, однако, свойственно асинхронное взаимодействие, при котором используются два отдельных генератора, как показано на рис. 5.4. Сообщение разбивается на символы (обычно длиной 5–8 бит), и два генератора синхронизируются в начале передачи каждого символа.

Состоянию не занятости линии соответствует сигнал 1 (в сетях связи называемый «меткой»). Передача символа начинается с сигнала 0 (называемого паузой), за которым следуют биты данных, и обычно первым передается самый младший бит. Иногда после би-



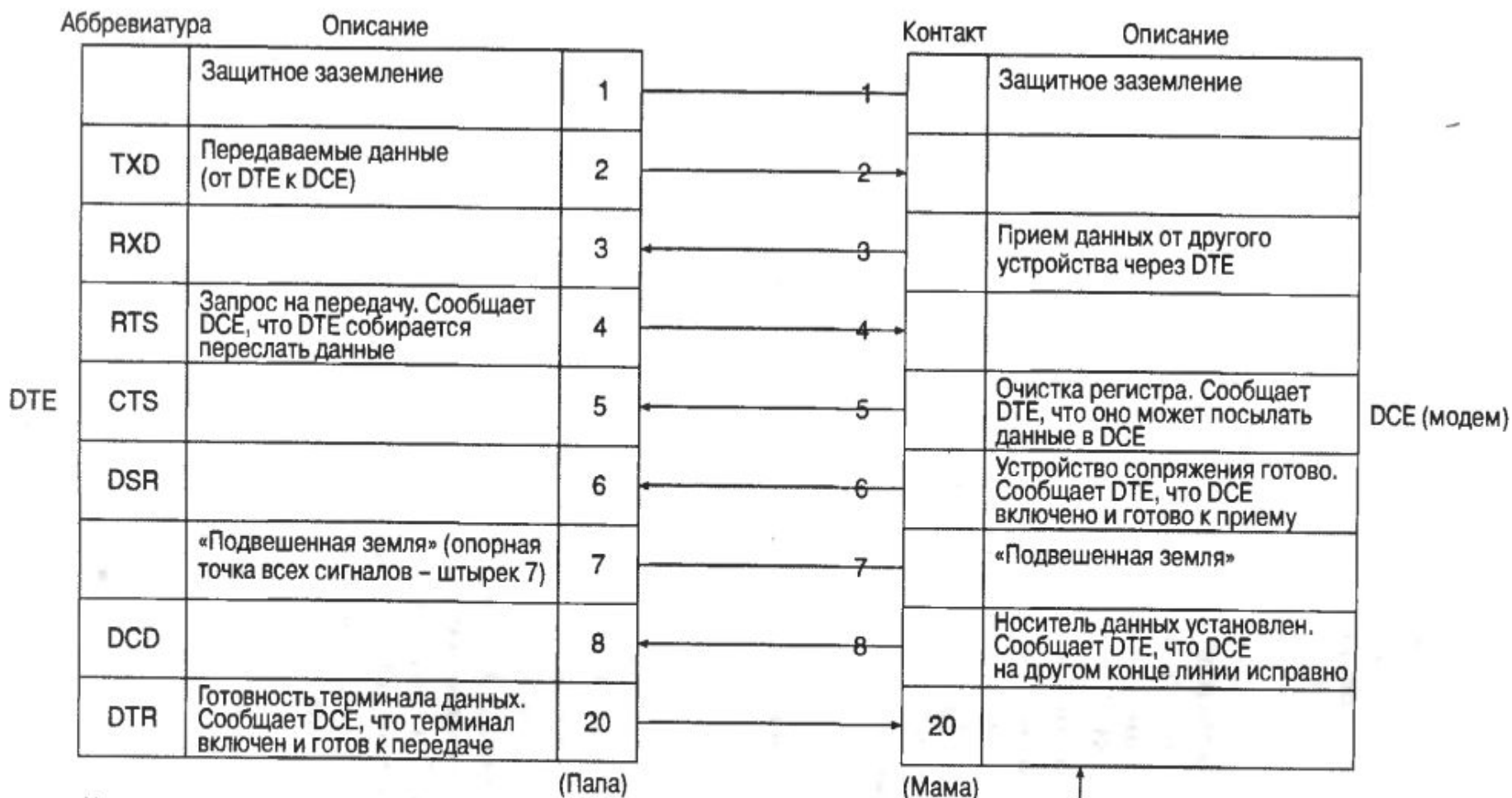
## *RS232E*

Для успешного осуществления связи должен существовать набор правил, в соответствии с которыми производится передача данных. Эти правила можно разделить на стандарты, которые устанавливают уровни напряжений, соединение и управление взаимодействием между DTE и DCE, и протоколы, определяющие содержание и контроль собственно сообщений.

Многие из первых работ по передаче данных были выполнены в США компанией Bell Telephone, и их результат был оформлен Ассоциацией отраслей электронной промышленности (EIA) как «Стандарт взаимодействия между DTE и DCE, использующий последовательный обмен данными в виде двоичного кода». Этот стандарт известен как RS232, в настоящее время распространена его версия E.

Общемировые стандарты установлены организацией Comite Consultatif International Telephonique et Telegraphique (CCITT), являющейся частью Международного телеграфного союза ООН. CCITT публикует стандарты и рекомендации, которые для передачи данных имеют префиксы V или X. Для всех практических применений стандарт V24 идентичен RS232.

Уровни сигналов, установленные стандартами RS232 и V24, составляют от +6 В до +12 В на передающей стороне для паузы (ноль) и от -6 В до -12 В для метки (единица). На приемной стороне допускается уменьшение этих сигналов до +3 В и -3 В. Кроме того, устанавливаются также другие характеристики, такие как пропускная способность линии связи и скорость нарастания фронта сигналов. Соединения выполняются при помощи 25-штырькового разъема D-типа.



Номера контактов соответствуют 25-штырьковому разъему D-типа. В системе минимальной конфигурации используются контакты 2, 3, 7. Спецификация определяет и другие сигналы для дополнительного канала, вызова и т. д.



**Рис. 5.8.** Соединения между DTE и DCE в спецификации RS232



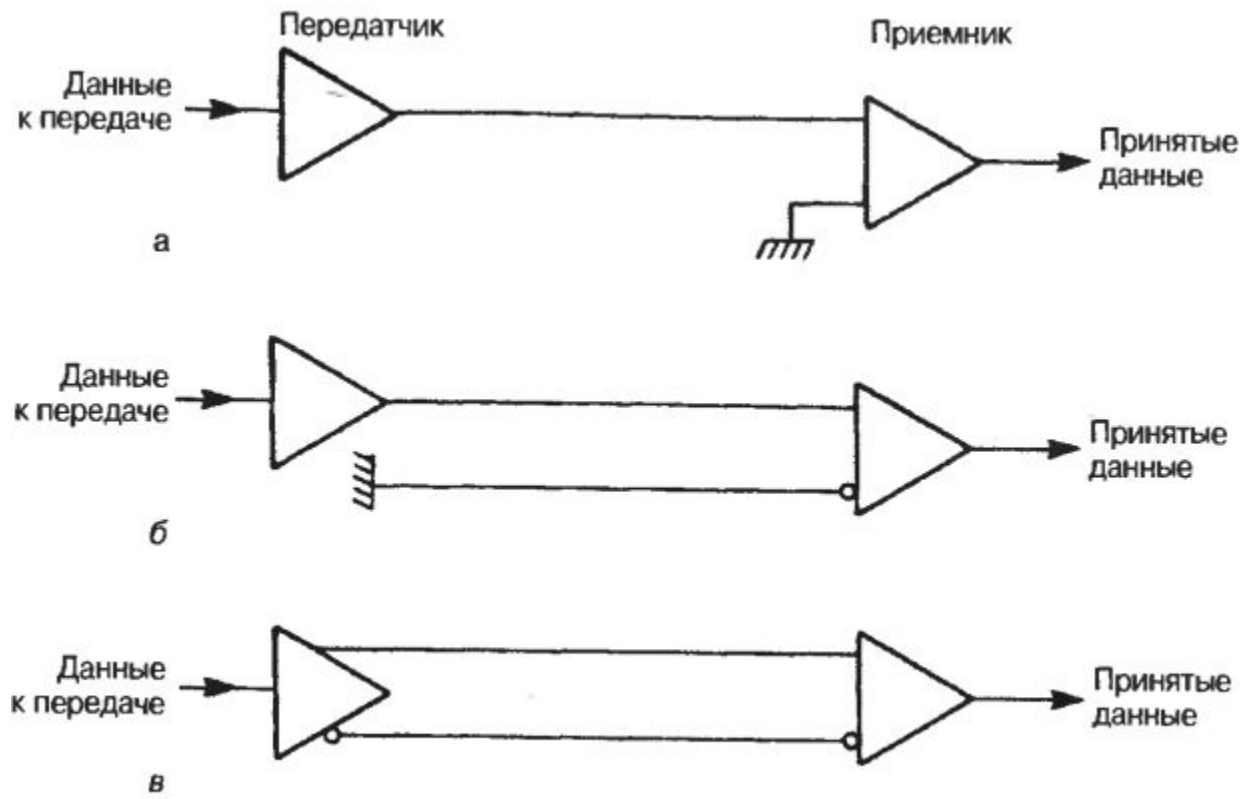
### *RS422 и RS423*

Стандарт RS232 был разработан для связи между DTE и DCE небольшой протяженности, обычно в пределах одной и той же комнаты. Если использовать RS232 для высокоскоростной передачи данных на большие расстояния (более нескольких метров), то здесь будут возникать проблемы.

Ассоциация отраслей электронной промышленности (EIA), признавая ограниченные возможности RS232 для связи между DTE и DCE, разработала и выпустила два других стандарта, проиллюстрированные на рис. 5.9. Одной из основных проблем, связанных с RS232, является отсчет всех сигналов относительно «подвешенной земли» (контакт 7 на рис. 5.8), как показано на рис. 5.9 (а). В RS423 и RS422 (рис. 5.9 (б) и (в)) применяются дифференциальные приемники, позволяющие обойтись без общего заземления и скомпенсировать влияние помех.

В передатчике номинальные значения напряжений установлены в виде  $\pm 6$  В, а смысл сигнала определяется его относительной полярностью. Для метки (единица) контакт А имеет отрицательный потенциал относительно В, а для паузы – наоборот.

В RS423 используются передатчик с несимметричным выходом и приемник с дифференциальным входом, что позволяет в случае применения передатчика, работающего в стандарте RS232, обеспечить разность потенциалов земли, не превышающую 4 В. В RS422

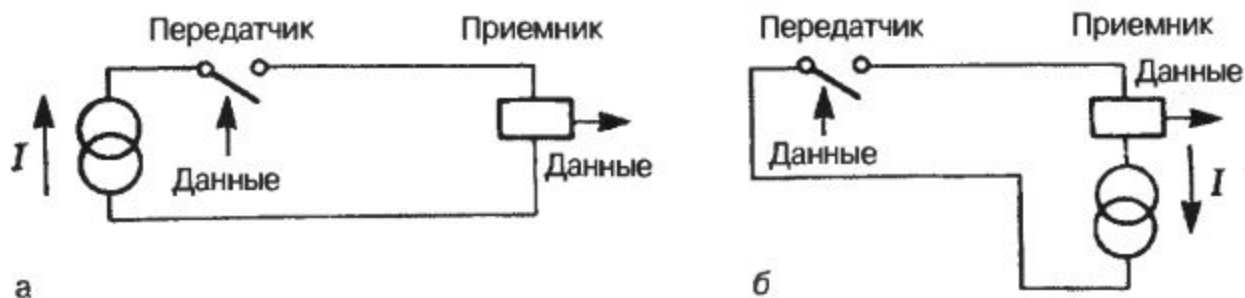


**Рис. 5.9.** Стандарты передачи данных: (а) RS232; (б) RS423; (в) RS422

### Токовая петля 20 мА

Неофициальным ранним «стандартом» является токовая петля 20 мА. Его происхождение связано с распространенными прежде электро-механическими телетайпами, но до сих пор этот стандарт еще можно встретить во многих приложениях. Стандарт включал в себя ключ, управляемый данными в передатчике, источник тока и датчик тока на приемной стороне. Наличие тока в цепи соответствует метке (единица), а отсутствие — паузе (ноль).

Токовая петля, изолированная от земли, обеспечивает невосприимчивость к помехам и игнорирование различий в потенциалах земли на обоих концах петли. Это основная причина, по которой продолжается ее практическое использование.



**Рис. 5.10.** Два способа передачи данных по токовой петле 20 мА:

- (а) активный передатчик, пассивный приемник;
- (б) пассивный передатчик, активный приемник

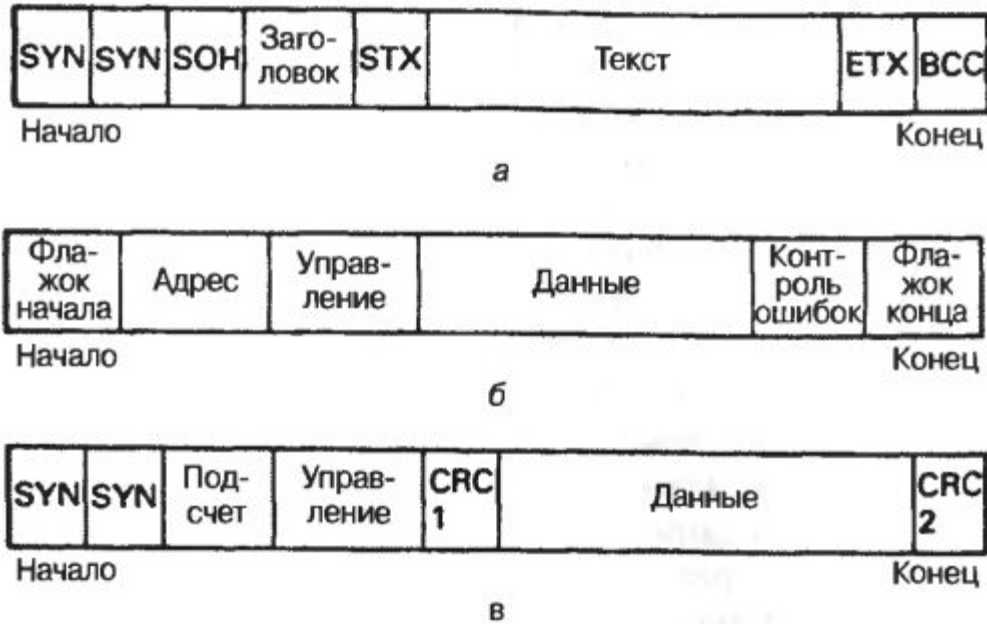
### *Протоколы сообщений*

Описанные выше стандарты имеют отношение к «механике» передачи данных, а содержание сообщения определяется используемым протоколом. Кроме определения формы сообщения (т. е. какая группа битов образует символы и какие группы символов образуют сообщение) протокол должен устанавливать, как иницируется и завершается связь и какие необходимо предпринять действия, если связь прерывается во время передачи сообщения. Протокол также должен определять, как обнаруживаются ошибки и какие за этим следуют действия.

По существу получили распространение три типа протоколов, как показано на рис. 5.11. Знак-ориентированные протоколы (рис. 5.11 (а)) для форматирования сообщения используют управляющие символы из набора ASCII (см. табл. 5.1). В некоторой степени большинство таких протоколов основаны на стандарте BISYNC корпорации IBM.

Бит-ориентированные протоколы, такие как SDLC корпорации IBM и HDLC и CCITT X25 Международной Организации по стандартизации, построены по схеме, приведенной на рис. 5.11 (б). Начало и конец сообщения определяются флажками признака, причем флажку конца предшествует контроль на наличие ошибок.

Последним типом протоколов является протокол с подсчетом байтов. Начало сообщения отмечается флажком, за которым следует подсчет, показывающий общее количество символов в сообщении. Приемник подсчитывает символы в сообщении и после проверки на наличие ошибок подтверждает получение. Из протоколов этого типа наибольшее распространение имеет DDCMP фирмы DEC.



**Рис. 5.11.** Три типа протоколов, обычно используемых при передаче данных: (а) основной вид знак-ориентированного протокола; (б) основной вид бит-ориентированного протокола; (в) основной вид протокола с подсчетом байтов. Обратите внимание на двукратный контроль на наличие ошибок – один для заголовка (подсчет и управление) и один для данных

### Контроль ошибок

Наложение шума на дискретный сигнал не обязательно приводит к искажению информации. На приемной стороне исходный сигнал может быть восстановлен, если шум не настолько интенсивен, чтобы превратить 1 в 0 и наоборот.

Шум обычно имеет распределение спектральной плотности мощности, приведенное на рис. 5.12, с нулевым средним значением и «хвостами», простирающимися в обе стороны до бесконечности. Если уровни напряжения дискретного сигнала имеют значения  $+V$  и 0, то в области А шум будет превращать 0 в 1, а шум в области В — 1 в 0. Таким образом, вероятность ошибки равна сумме площадей А и В, деленной на общую площадь под кривой распределения.



**Рис. 5.12.** Спектральная плотность мощности шума.

Шум в областях А и В будет приводить к искажениям дискретного сигнала напряжением  $V$  вольт между значениями 1 и 0

Существуют два способа действий при наличии ошибок. Самый простой, используемый почти во всех промышленных системах, заключается в том, что при обнаружении ошибки приемное устройство запрашивает повторение передачи. Этот метод известен под названием автоматический запрос повторной передачи или ARQ (automatic transmission on request). Для квитирования и контроля здесь используются упомянутые в разд. 5.2.6 символы ACK и NAK из набора ASCII.

Второй способ основан на обнаружении и исправлении любых ошибок путем добавления в сообщение избыточных символов. Этот метод носит название опережающего контроля ошибок или FEC (forward error control). Английский язык обладает значительной избыточностью (что позволяет обмениваться речевыми сообщениями в трудных условиях). Если мы имеем предложение:

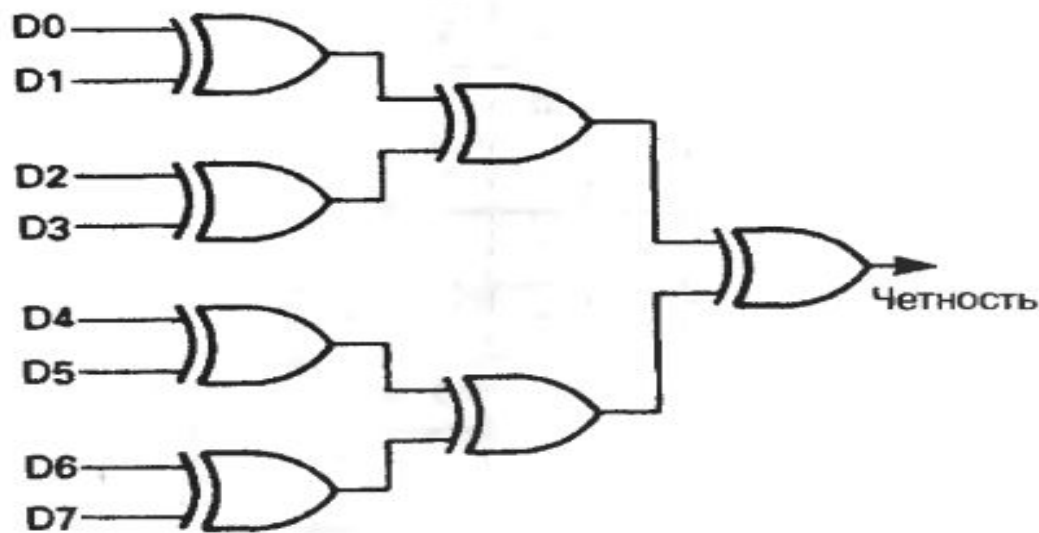
Today the weather is sunny,

в котором 40 % ошибок, и знаем, что речь идет о погоде, то нетрудно заполнить пропущенные позиции символами, чтобы получить «Today the weather is sunny».

Простейшим способом обнаружения ошибки является использование контрольного бита четности. Это дополнительный бит, с помощью которого гарантируется, что количество битов в одном символе или байте всегда нечетное, как показано на рис. 5.14 (а). Это называется проверкой на нечетность; можно осуществить также проверку на четность (контрольный бит добавляется, чтобы сделать количество битов в каждом символе четным), но проверка на нечетность используется гораздо чаще. В коде ASCII символ содержит 7 битов, поэтому добавление бита четности увеличивает длину до 8 битов.

Для 8-битового символа контроль четности легко осуществляется с помощью вентиля «исключающее ИЛИ», как показано на рис. 5.14 (б). Существуют специальные интегральные схемы для определения четности, такие как ТТЛ-схема 74180 и КМОП-схема 4531.

Контроль по четности (или, используя его полное название, поразрядный контроль четности) может обнаружить один (или 3, 5, 7) ошибочный бит, но будет нарушен при четном количестве (2, 4, 6) ошибочных битов.

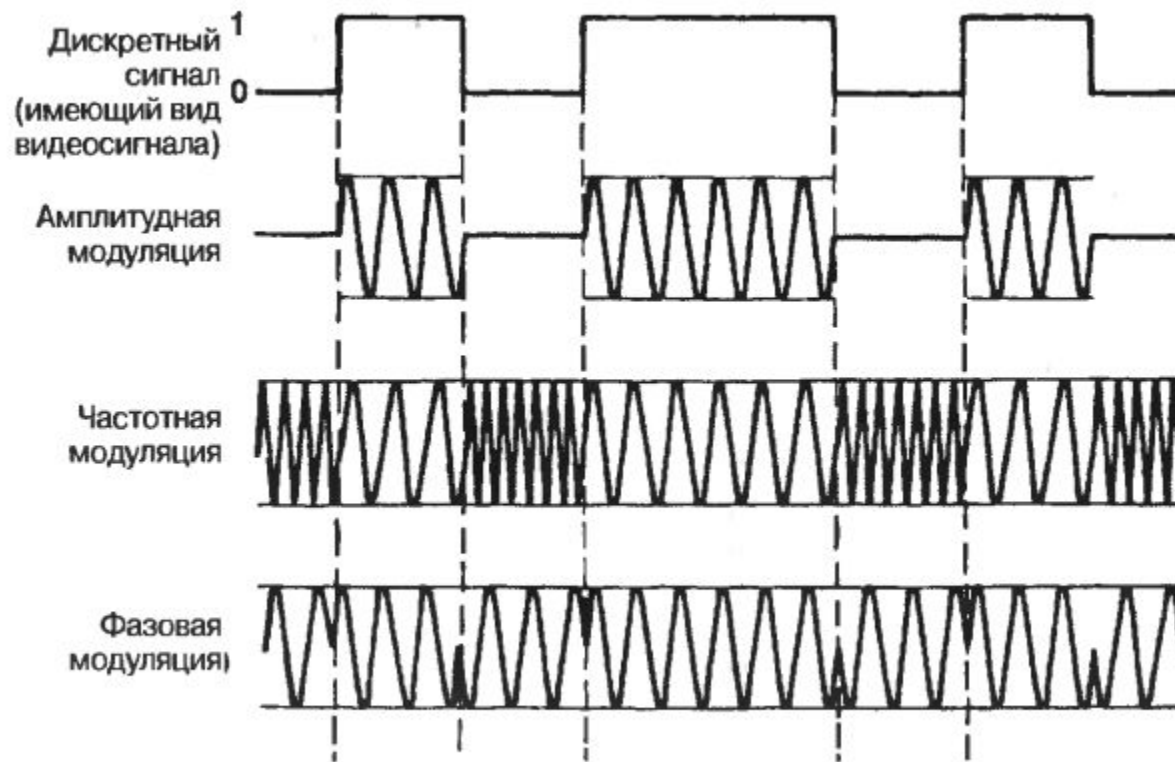




## **Модуляция дискретных сигналов**

До сих пор мы считали, что при последовательной передаче данных цифровые сигналы имеют «сырой» вид, т. е. представляют собой ряд уровней напряжения, соответствующих определенной комбинации битов. Такой способ называется прямой (немодулированной) передачей.

Дискретный сигнал занимает полосу частот от 0 Гц (сигнал постоянного тока, соответствующий сплошной последовательности нулей или единиц) до почти половины частоты передачи битов. Многие средства связи, такие как радиоканалы и телефонные сети, объективно имеют ограничения на передачу низкочастотных сигналов и поэтому не могут непосредственно работать с видеосигналами. По этой причине дискретные данные путем модуляции преобразуются в колебания несущей частоты. Существуют три разных способа модуляции: амплитудная модуляция (АМ), частотная модуляция (ЧМ) и фазовая модуляция (ФМ). Все они проиллюстрированы на рис. 5.6. Одним из преимуществ модуляции является то, что несколько независимых сигналов, преобразованных в колебания различных несущих частот, могут быть переданы по одной и той же линии связи. В этом случае говорят, что модулированный дискретный сигнал занимает широкую полосу частот или передает-



**Рис. 5.6.** Различные виды модуляции дискретных сигналов