

3. Взаимодействие устройств типа DTE/DCE в синхронном режиме: нестандартные решения

Рассмотренные в этой главе нестандартные решения позволяют улучшить характеристики существующих систем передачи данных. Приведем краткий обзор этих решений.

В схеме передачи данных, рассмотренной в п. 3.1, байты или иные группы битов разграничены стартовыми и стоповыми битами, и в то же время применен синхросигнал. В результате получена более надежная временная диаграмма передачи данных, с явным указанием границ и центров битовых интервалов.

В следующей схеме (п. 3.2) одна и та же линия используется как для передачи синхросигнала, так и для передачи временных меток для разграничения информационных кадров. Это позволяет получить простую и надежную систему передачи данных с битовой и кадровой синхронизацией.

Группа решений по п. 3.3 основана на усовершенствовании классического способа передачи данных с использованием сигналов TxD и CLK. Как уже отмечалось, здесь модернизируется простейшая схема, считавшаяся безупречной на протяжении десятилетий! В результате расширены функциональные возможности систем передачи данных и повышена скорость обмена данными.

Классическое аппаратное управление потоком данных между устройствами типа DTE и DCE подразумевает обмен сигналами RTS – CTS (см. п. 1.2). Как показано в п. 3.4, можно исключить эти сигналы и упростить схемы синхронизации, если вместо непрерывного потока синхросигналов использовать пачки импульсов.

Решения, рассмотренные в п. 3.5, позволяют дистанционно регулировать фазу сигнала от удаленного синхрогенератора. Это, в частности, дает возможность передавать синхросигналы “против течения” потока данных, что на первый взгляд представляется невозможным! Показано использование такой передачи при построении удлинителя несинхронных сигналов TxS и RxS.

В п. 3.6 рассмотрена экономичная по аппаратуре система с непосредственной передачей синхросигнала по каналу связи. Она применима, в частности, для дуплексной передачи данных с синхронизацией от общего источника синхросигналов TxS и RxS.

Задача распознавания начала сообщения в асинхронном потоке данных рассмотрена в п. 3.7. Обычно сообщения разделяются паузами, но они могут возникнуть в результате “замираний” сигнала или вследствие действия помех. Чтобы отличить истинную разделительную паузу от ложной, в предлагаемом решении применяется простой способ кодирования символов.

При передаче непрерывного асинхронного потока данных служебные биты Старт и Стоп “прячутся” среди информационных битов. Как в этом случае передатчик может “помочь” приемнику в отыскании служебных битов? Ответ на этот вопрос можно найти в п. 3.8. В заключение главы (п. 3.9) рассмотрены решения задачи распознавания межбайтовых границ в непрерывном синхронном потоке данных.

3.1. Синхронная передача данных между устройствами типа DTE и DCE с использованием стартовых и стоповых битов

В рассмотренном далее решении применена комбинация идей синхронного и асинхронного обмена данными. Далее для упрощения изложения будет рассмотрен только односторонний обмен данными, но все сказанное применимо также и к передаче данных во встречном направлении. Напомним, в чем заключается асинхронный обмен (рис. 3.1).

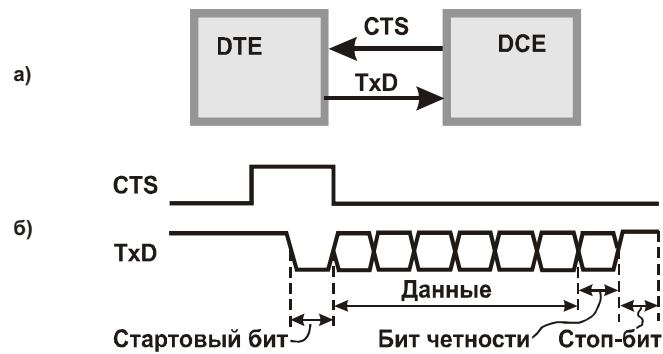


Рис. 3.1. Передача данных в асинхронном режиме:
а – типовая схема; *б* – временная диаграмма

При обнаружении сигнала готовности устройства DCE к приему данных (положительного напряжения в линии CTS) устройство DTE, если это необходимо, начинает передачу асинхронной посылки. Она состоит из стартового бита, семи или восьми битов данных (для упрощения рисунка показаны только шесть битов) и может сопровождаться битом контроля на четность (или нечетность; все параметры посылки, конечно, однозначно задаются до проведения сеанса связи). И, наконец, в завершение передаются один или два стоп-бита, которые служат разделителями посылок, если они следуют непрерывным потоком.

Устройство DCE обнаруживает отрицательный фронт сигнала TxD (начало стартового бита) и начинает отсчет времени с целью определения моментов, соответствующих серединам битовых интервалов. В эти моменты данные записываются в приемный регистр устройства DCE.

Надежность такого способа обмена снижается по мере повышения скорости передачи данных. Импульсы теряют прямоугольную форму из-за того, что битовый интервал становится соизмеримым с длительностью фронта сигнала. Точность построения синхросетки устройством DCE снижается, так как фронт стартового бита становится все более пологим; кроме того, становится ощутимым джиттер (дрожание фазы – см. гл. 5) сигнала TxD.

Синхронный обмен обеспечивает более высокую скорость передачи данных. Это достигается как за счет исключения стартовых и стоп-битов, так и благодаря явному (а не подразумеваемому) подтверждению истинности каждого передаваемого бита данных сигналом CLK на дополнительно введенной линии синхронизации. Однако при синхронном обмене необходима упаковка данных в кадры определенного формата, чтобы устройство DCE могло иметь ориентиры в непрерывном входном потоке битов. Это устройство распознаёт границы и распаковывает кадры. Для одних приложений наиболее целесообразно использовать синхронную передачу данных, для других – асинхронную.

Предлагаемое в [13] решение (рис. 3.2) позволяет повысить надежность асинхронного обмена при высоких скоростях передачи данных. Это достигается благодаря тому, что, в отличие от классического асинхронного способа передачи данных, истинность передаваемых битов, как и при синхронной передаче, подтверждается отрицательными фронтами синхросигнала CLK, а границы битовых интервалов соответствуют положительным фронтам этого сигнала.

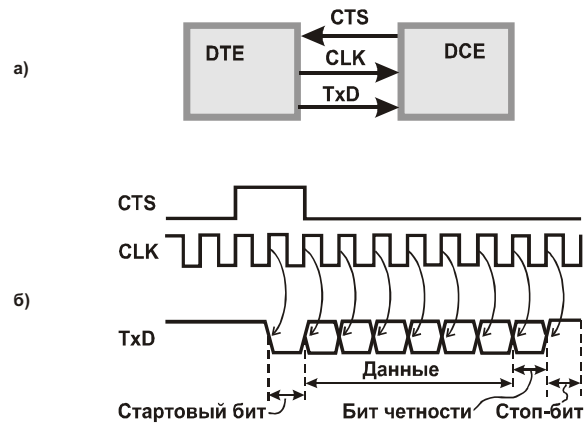


рис. 3.2. Передача данных в синхронном режиме: *а* – типовая схема; *б* – временная диаграмма, сочетающая принципы синхронного и асинхронного обмена

В исходном состоянии на вход устройства DCE поступает поток лог. 1, что соответствует отсутствию полезных данных. При наличии готовности принять данные устройство DCE устанавливает положительное напряжение в линии CTS (в данном примере имеется готовность принять только одну посылку). Устройство DTE, обнаружив готовность, формирует и выдает посылку, размещая ее в синхросетке сигнала CLK.

Устройство DCE расценивает первый принятый нулевой бит как стартовый и снимает сигнал готовности приема (это касается только данного примера – сигнал готовности может сохраняться в течение любого нужного периода). Стартовый и последующие биты принимаются по отрицательным фронтам синхросигнала CLK. Таким образом, сохраняется асинхронный способ передачи данных, при котором посылки передаются с произвольным темпом (вплотную или с любыми интервалами). В то же время внутри каждой посылки обеспечивается жесткая привязка битов к синхросигналу, что повышает надежность приема.

Описанный способ обмена данными применим также к комбинациям устройств DTE – DTE и DCE – DCE.

3.2. Синхронный обмен данными с передачей кадровых меток

Неструктурированный поток данных (без начала, конца и каких-либо промежуточных “опознавательных знаков”) может представлять ограниченный интерес и используется разве что при отладке или поиске неисправностей системы на физическом уровне. При работе транспортных систем передаваемые данные объединяются в логически законченные структурные единицы: кадры, пакеты и проч. Эти единицы (для определенности – кадры) могут значительно различаться по форме и содержанию в зависимости от принятой технологии передачи данных, протоколов обмена и т. д. Однако их объединяет то, что в явном или неявном виде должны быть заданы границы, по которым приемник может выделить кадр из потока.

В простейшем случае задача разграничения кадров решается аппаратно: между передатчиком и приемником вводится специальная линия, по которой передаются импульсы – кадровые метки. В примере, приведенном на рис. 3.3, передатчик взаимодействует с приемником в синхронном режиме и посылает ему кадровые метки FR для обозначения межкадровых границ. Границы битовых интервалов соответствуют положительным фронтам синхросигнала CL. Прием данных происходит по отрицательным фронтам этого синхросигнала.

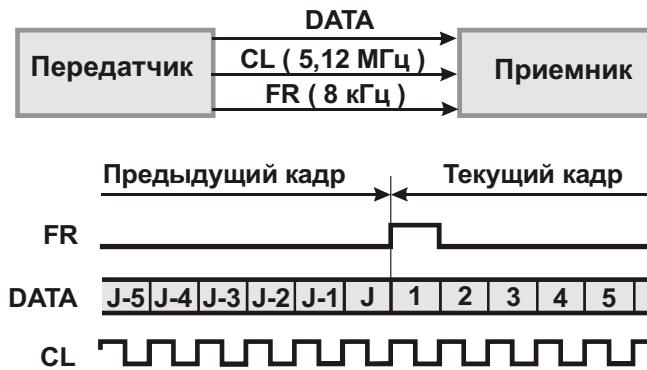


рис. 3.3. Раздельная передача данных DATA, синхросигнала CL и кадровых меток FR

Кадры следуют с частотой 8 кГц. Каждый кадр содержит J битов; в данном примере $J = 640$. Скорость передачи данных составляет $640 \times 8 = 5120$ бит/с. Кадровая метка совпадает во времени с первым битовым интервалом кадра.

Чтобы уменьшить число линий канала связи, сохраняя передачу кадровых меток, применяют разные методы кодирования. Например, используют только одну линию, по которой передается манчестерский код (см. п. 8.1.4). В этом коде сигналы DATA и CL перед выдачей в линию суммируются по модулю два. Кадровые метки отображаются моментами преднамеренного нарушения правил кодирования [61].

В схеме, приведенной на рис. 3.4 [60], число линий канала связи уменьшено до двух.

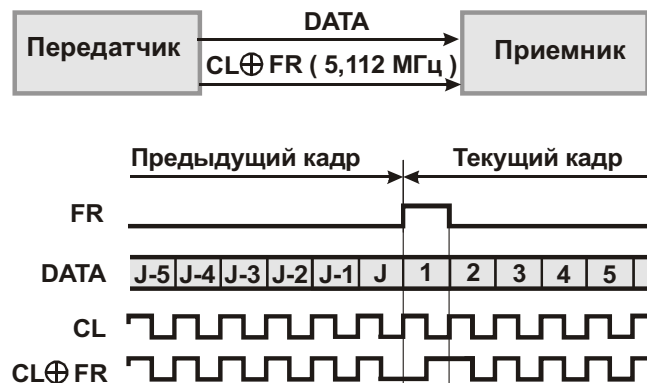


рис. 3.4. Передача данных DATA и суммарного синхросигнала $CL \oplus FR$

Сигнал общей синхронизации $CL \oplus FR$ получен суммированием по модулю 2 битовых CL и кадровых FR синхроимпульсов. Из-за периодического прореживания синхроимпульсов частота сигнала $CL \oplus FR$ на 8 кГц ниже частоты сигнала CL и составляет 5,112 МГц.

Схема восстановления синхросигналов и временные диаграммы приведены на рис. 3.5. Схема содержит генератор синхросигнала частотой 184,32 МГц, в 36 раз превышающей частоту сигнала CL передатчика. Стабильность генератора обеспечивается петлей фазовой автоподстройки частоты. Эта петля предназначена для привязки формируемых сигналов к входному синхросигналу, точнее, к его основной составляющей CL частотой 5,12 МГц. Фазовый компаратор следит за совпадением фронтов входных сигналов. При обнаружении более или менее устойчивого разбаланса фронтов корректируется напряжение U управления частотой генератора в направлении улучшения совпадения. Инерционность петли управления определяется частотой среза фильтра низких частот. Благодаря инерционности, устройство не реагирует на кратковременные помехи и пропадания входного сигнала. Обнаружение начала кадра сопровождается появлением сигнала $FRAME = 1$.

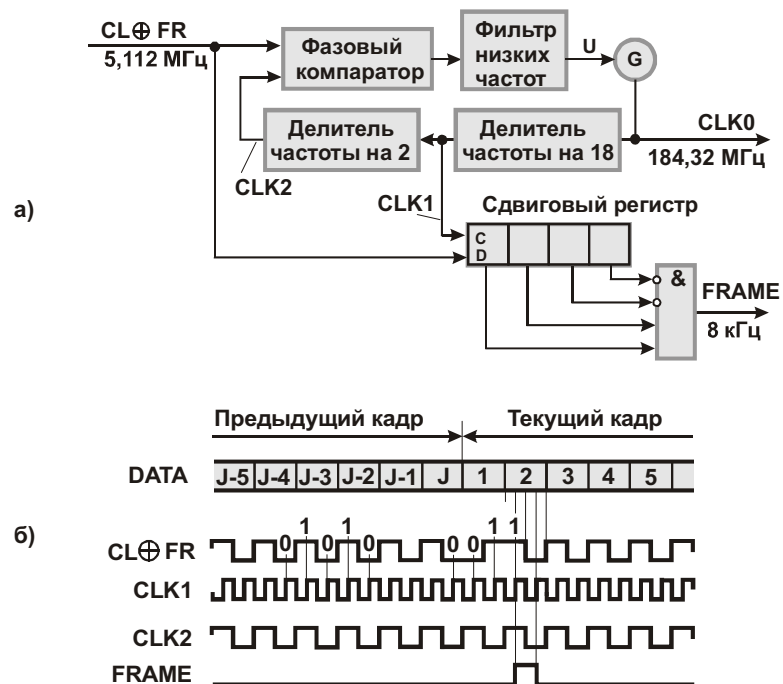


рис. 3.5. Восстановление синхросигналов в приемнике: *а* – схемное решение; *б* – временные диаграммы

Сигнал $CL \oplus FR$ непрерывно анализируется схемой на основе сдвигового регистра и элемента И. Сдвиговый регистр продвигает информацию слева направо по положительным фронтам сигнала $CLK1$. Элемент И содержит по два неинвертирующих и инвертирующих входа. Как следует из временных диаграмм, при передаче битов, далеких от границ кадров, в сдвиговом регистре присутствует код 0101 или 1010. При этом на выходе элемента И сформирован сигнал $FRAME = 0$. (Возможными кратковременными всплесками сигнала $FRAME$ в моменты переходных процессов в сдвиговом регистре пренебрегаем, так как опрос этого сигнала производится последующими схемами, например по отрицательному фронту сигнала $CLK1$, когда переходные процессы отсутствуют.)

При прохождении межкадровой границы в сдвиговом регистре присутствует код 1100, на выходе элемента И формируется сигнал $FRAME = 1$. Длительность этого сигнала равна половине битового интервала, но его положение, как следует из рис. 3.5, б, не вполне соответствует межкадровой границе. Учесть соответствующую поправку несложно, тем более, что в нашем распоряжении имеется ряд опорных частот, вплоть до частоты, в 36 раз превышающей скорость передачи данных. Наличие столь мелкой временной сетки позволяет вести подробную “прорисовку” сигналов, адаптироваться к их искажениям, выделять помехи, выбирать наиболее надежные точки опроса данных и т. п.

3.3. Повышение быстродействия и расширение функциональных возможностей схемы с попутной синхронизацией

В гл. 1 были рассмотрены два способа синхронного обмена данными между устройствами. Рассмотренные далее усовершенствования касаются первого способа, который предусматривает попутную передачу двух сигналов TxD и CLK (соответственно сигнала данных и синхросигнала), что поясняется рис. 3.6.

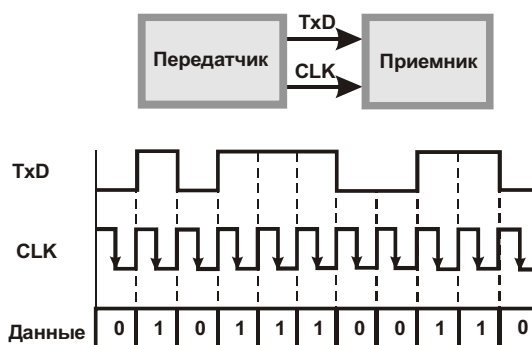


Рис. 3.6. Классическая временная диаграмма передачи данных в синхронном режиме

Положительные фронты синхросигнала CLK задают границы между битовыми интервалами; эти интервалы обозначены на рисунке штриховыми линиями. Данные принимаются по отрицательным фронтам синхросигнала.

Эта схема успешно применяется не один десяток лет, и лишь сравнительно недавно (в 2000 и 2001 годах) предложены рассмотренные далее варианты ее кардинального улучшения [11], [63].

3.3.1. Удвоение скорости передачи данных с использованием положительного и отрицательного фронтов сигнала CLK для их приема

В первом варианте усовершенствованной схемы для приема данных используются как положительные, так и отрицательные фронты синхросигнала CLK (рис. 3.7) [63]. Сравнивая временные диаграммы, приведенные на рис. 3.6 и рис. 3.7, можно отметить, что при одинаковой скорости передачи данных частота сигнала CLK в усовершенствованной схеме уменьшается вдвое. Можно дать и иную трактовку тех же нововведений: при неизменной частоте сигнала CLK скорость передачи данных удваивается.

Разумеется, оба фронта сигнала CLK должны соответствовать установленным значениям сигнала TxD, точнее, должны обеспечиваться достаточные интервалы предустановки и удержания сигнала TxD соответственно до и после прохождения фронта сигнала CLK. Эти параметры определяются техническими данными применяемой элементной базы и параметрами линии связи.

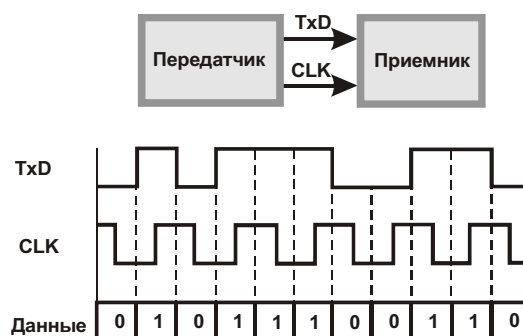


Рис. 3.7. Усовершенствованная временная диаграмма передачи данных в синхронном режиме – первый вариант

Примечательно, что усовершенствованная схема совместима с традиционным оборудованием приема данных, работающим в соответствии с предыдущей временной диаграммой. Действительно, теперь поток передаваемых данных содержит две составляющие, одна из которых удовлетворяет общепринятой временной диаграмме (см. рис. 3.6), а вторая представляет собой некое дополнение. Это дополнение просто не воспринимается традиционным оборудованием. Поэтому традиционная и новая аппаратура

приема данных могут включаться параллельно. Однако передатчик должен знать, с каким приемником ему предстоит работать в данном сеансе связи.

Рассмотрим пример воплощения рассмотренной идеи. В схеме передачи данных, показанной на рис. 3.8, параллельные коды DATA 1 и DATA 2 могут относиться к одному или двум независимым информационным потокам. Эти потоки поступают в передатчик от внешнего источника (на рисунке не показан) и суммируются в линии TxD. Первый воспринимается приемником по отрицательным фронтам сигнала CLK, второй – по его положительным фронтам.

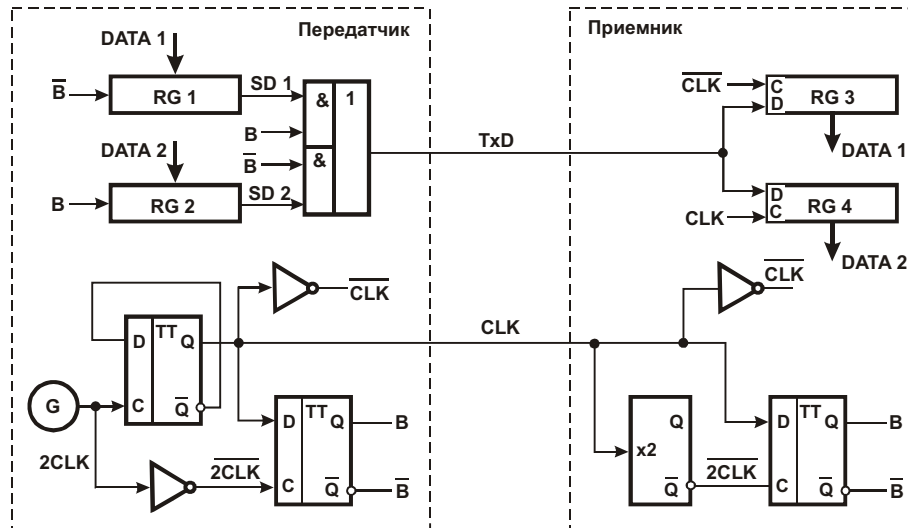


рис. 3.8. Схема передачи данных. Приемник получает биты данных TxD как по положительным, так и по отрицательным фронтам сигнала CLK

Передатчик содержит генератор G, сдвиговые регистры RG 1 и RG 2, элемент 2И-ИЛИ, два D-триггера и два инвертора. Приемник содержит сдвиговые регистры RG 3 и RG 4, D-триггер, инвертор и удвоитель "x2" частоты с инверсным выходом.

Генератор G формирует непрерывную последовательность синхроимпульсов 2CLK с частотой, равной скорости передачи данных TxD по линии (рис. 3.9).

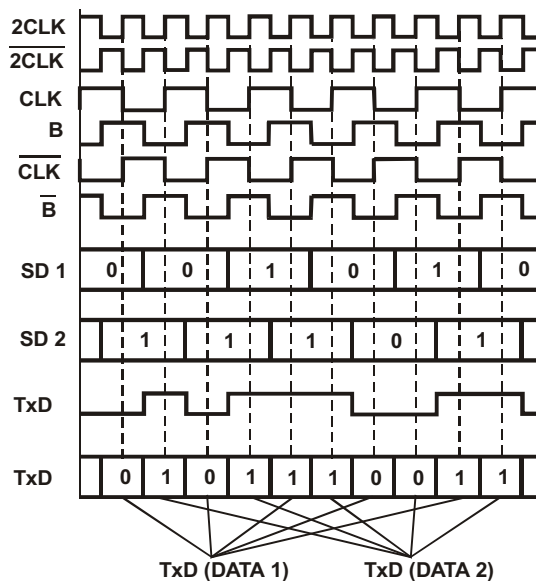


рис. 3.9. Временные диаграммы передачи данных в схеме, показанной на рис. 3.8

Сигнал с генератора G преобразуется в четыре синхросигнала с последовательным фазовым смещением на четверть периода. Данные DATA 1 и DATA 2 периодически заносятся в регистры RG 1, RG 2 и затем сдвигаются в них по положительным фронтам соответствующих управляющих сигналов (цепи управления параллельной записью данных в регистры RG 1 и RG 2 на рисунке не показаны).

Последовательные данные SD 1 и SD 2, выдвигаемые из регистров RG 1 и RG 2, поочередно выдаются в линию через логический элемент 2И-ИЛИ. Поэтому передаваемые данные TxD содержат биты TxD (DATA 1) и TxD (DATA 2), принадлежащие двум разным или одному общему информационным потокам.

Приемник последовательно загружает принимаемые данные в сдвиговые регистры RG 3 и RG 4 с противофазной синхронизацией. При этом информация распределяется по двум первоначальным направлениям. На выходах регистров в соответствующие моменты формируются параллельные данные DATA 1 и DATA 2. В эти моменты они считываются последующими устройствами (эти устройства и схемы управления считыванием на рисунке не показаны).

Удвоитель "x2" частоты сигнала CLK и D-триггер позволяют получить в приемнике полный комплект аналогов синхросигналов передатчика, если в этом есть необходимость.

3.3.2. Удвоение скорости передачи данных заменой сигнала CLK сигналом разграничения одноименных битов

Рассмотрим второй вариант усовершенствования классической схемы передачи данных. Как показано на рис. 3.10, вместо периодического сигнала CLK (см. рис. 3.6) применен сигнал разграничения DELIM, который изменяется в тех случаях, когда передаваемые данные TxD неизменны и остается неизменным, когда данные изменяются [11]. Граница очередного битового интервала определяется приемником по факту изменения состояния пары принимаемых сигналов (TxD и DELIM).

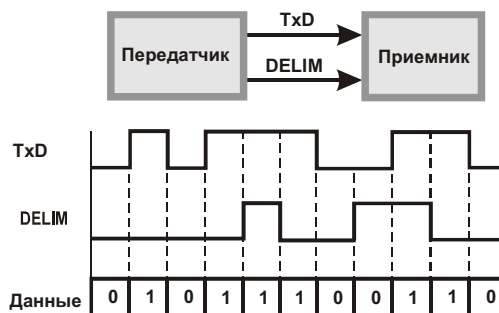


Рис. 3.10. Усовершенствованная временная диаграмма передачи данных – второй вариант

Сравнивая временные диаграммы, приведенные на рис. 3.6 и рис. 3.10, можно отметить следующее.

Во-первых, передаваемые данные формируются с одинаковым разбиением сигнала TxD на битовые интервалы. Во-вторых, сигнал DELIM в усовершенствованной схеме претерпевает всего лишь четыре изменения, в то время как сигнал CLK изменяется 21 раз. В течение одного битового интервала сигнал CLK изменяется дважды, а сигнал DELIM – не более одного раза.

Таким образом, при сохранении физических параметров системы (длины кабеля, элементной базы) можно удвоить скорость передачи данных или при той же скорости передачи, что и в традиционной схеме, вдвое уменьшить максимально возможную частоту сигнала, сопровождающего данные. Кроме того, при передаче случайных данных сигнал DELIM, в отличие от сигнала CLK, не содержит ярко выраженных спектраль-

ных составляющих, что способствует снижению уровня излучаемых в линию помех. (Для преобразования данных к виду, близкому случайной последовательности битов, используют скремблеры, см. п. 8.4.)

3.3.3. Расширение функциональных возможностей системы с разграничением одноименных битов

Идея построения дополнительного канала связи

В схеме, показанной на рис. 3.10, данные транслируются по одной линии, в то время как вторая используется только для восполнения “пробелов” синхронизации. Нельзя ли и по ней передавать данные? Возможность такой передачи подтверждается рис. 3.11.

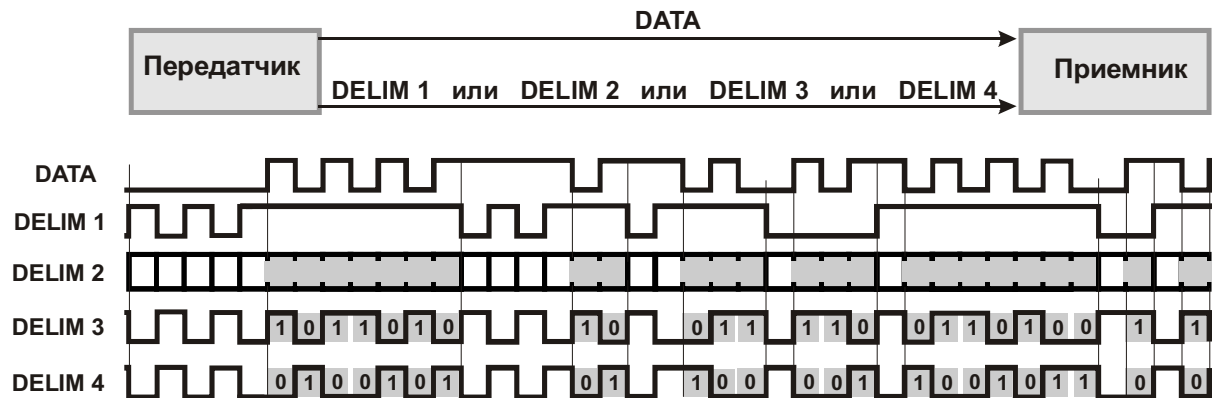


рис. 3.11. Временные диаграммы передачи данных с использованием сигналов DATA и разных вариантов построения сигнала DELIM: DELIM 1 и DELIM 2 – классический и обобщенный варианты; DELIM 3 и DELIM 4 – варианты передачи взаимно-обратных кодов

Сигнал DELIM 1 сформирован по рассмотренным ранее правилам, но можно их нарушить, увеличив информационную нагрузку на этот сигнал. Как показано в общем виде на диаграмме сигнала DELIM 2, существуют области, отмеченные серым фоном, в которых возможна любая последовательность сигналов лог. 0 и лог.1. Обязательные фронты сигнала DELIM 2 показаны в явном виде, необязательные – в форме зубцов. Отметим, что каждая затененная область начинается необязательным и заканчивается обязательным фронтом. Затененным областям соответствуют периоды “интенсивных” изменений сигнала DATA, поэтому эти изменения гарантируют распознавание битовых интервалов приемником независимо от состояния сигнала разграничения.

Временные диаграммы сигналов DELIM 3 и DELIM 4 отображают передачу кода 10110...1 и обратного ему 01001...0. В данном примере начальные участки диаграмм одинаковы, но в дальнейшем сигналы становятся противофазными.

Как следует из диаграмм, сигналы DELIM 1 – DELIM 4 в течение 15 битовых интервалов выполняют чисто административные функции, а в течение 24 битовых интервалов (отмеченных серым фоном) могут нести “полезную” информацию. По существу, сформирован дополнительный канал связи. Данные представлены двумя параллельными потоками – основным и дополнительным.

В приведенном примере данные имеют случайный характер; но если предположить, что сигнал DATA не изменяется, то сигналы DELIM 1 – DELIM 4 будут вынуждены изменяться в каждом такте, и дополнительный канал связи исчезнет.

Чтобы этого не произошло, поток данных DATA перед выдачей в линию скремблируется, т. е. определенным образом шифруется (см. п. 8.4). Шифрация заключается, например, в прогоне данных через сдвиговый регистр с элементами Иключающее ИЛИ в цепях обратной связи. При этом преобразование оказывается настолько запутанным, что любой входной поток данных становится на выходе псевдослучайным. Дешифрация (дескремблирование) потока выполняется приемником с помощью подобного сдвигового регистра с логическими элементами Иключающее ИЛИ. Далее предполагаем, что данные в линии связи скремблированы.

При равновероятном появлении сигналов лог. 0 и лог. 1 в потоке данных DATA пропускная способность дополнительного канала передачи данных составляет 50% пропускной способности основного канала. Действительно, вероятность перепада уровней сигнала DATA на границе битовых интервалов i и j равна 0,5 и не зависит от предыстории. Но именно эта вероятность определяет событие “вставки” очередного бита данных в поток сигналов разграничения битов. Таким образом, в среднем каждый второй битовый интервал пригоден для передачи дополнительного бита данных. В результате общая пропускная способность (или скорость передачи данных) системы по сравнению с прототипом (см. рис. 3.10) увеличена в 1,5 раза.

Рассмотренный способ создания дополнительного канала применим и к параллельной синхронной передаче данных по нескольким линиям. В схеме, показанной на рис. 3.12, данные передаются по трем линиям с общим сигналом DELIM разграничения битовых интервалов.

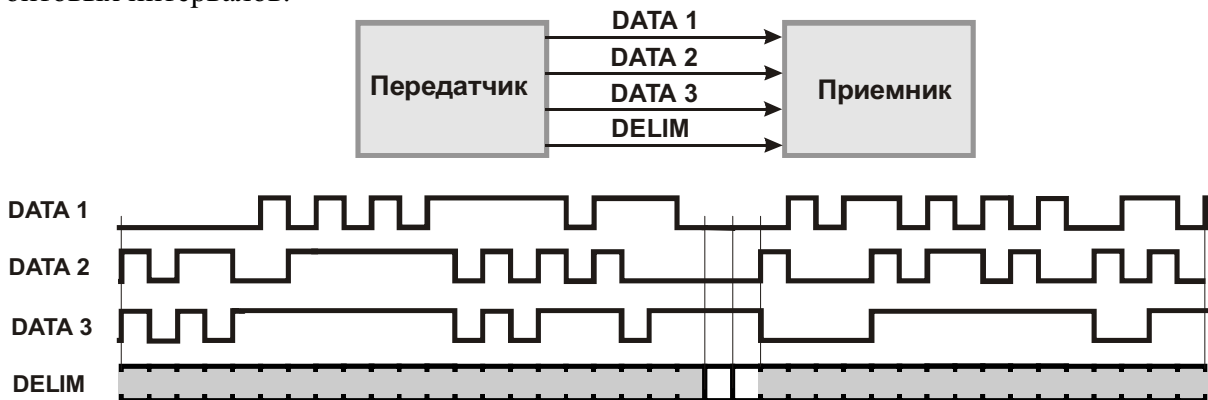


Рис. 3.12. Временные диаграммы передачи данных с использованием сигналов DATA 1 – DATA 3 и сигнала DELIM, показанного в обобщенном виде

Перепад сигнала в любой из четырех линий воспринимается приемником как момент разграничения битовых интервалов. Сигнал DELIM используется по прямому назначению только при одновременном отсутствии динамики сигналов DATA 1 – DATA 3. Вероятность таких событий невысока, поэтому затененная на рисунке область произвольных изменений сигнала DELIM более обширна по сравнению с аналогичной областью из предыдущего примера. Пропускная способность дополнительного канала связи приблизилась к пропускной способности каждого из трех основных каналов.

Для помехоустойчивого восстановления синхросигнала в приемнике может применяться генератор с фазовой автоподстройкой частоты. Примеры построения таких генераторов рассмотрены в гл. 9.

Принятые из четырех линий потоки данных записываются в память приемника и анализируются микропроцессором для разделения служебных и полезных битов сигнала DELIM. Такое разделение проводится по тому же алгоритму, который использовался при разметке приведенной на рисунке диаграммы этого сигнала. (При определенных комбинациях битов данных фрагменты диаграммы помечаются затененными областями и т. д.)

Пример схемной реализации системы с основным и дополнительными каналами связи

Рассмотрим пример воплощения описанной идеи построения дополнительного канала связи. Система передачи данных, показанная на рис. 3.13, в неявном виде содержит два канала передачи данных: основной DTE1 → DTE3 и дополнительный DTE2 → DTE4. Для передачи данных в обратном направлении необходима вторая пара устройств передатчик – приемник, включенная встречно.

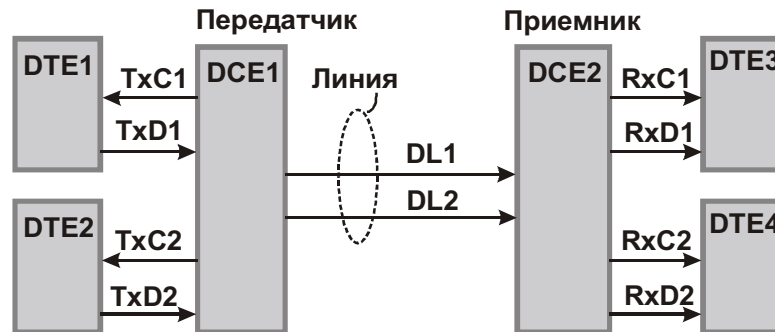


рис. 3.13. Система передачи данных с основным и дополнительными каналами связи

Пара сигналов TxС1 – TxD1 (TxС2 – TxD2) обеспечивает синхронную передачу данных из устройства DTE1 (DTE2) в передатчик DCE1. Положительные фронты сигнала TxС1 (TxС2) задают границы битовых интервалов. По отрицательным фронтам сигнала TxС1 (TxС2) соответствующий бит данных TxD1 (TxD2) временно запоминается в передатчике.

Пара сигналов RxC1 – RxD1 (RxC2 – RxD2) обеспечивает синхронную передачу данных из приемника DCE2 в устройство DTE3 (DTE4). Положительные фронты сигнала RxC1 (RxC2) задают моменты начала битовых интервалов. По отрицательным фронтам сигнала RxC1 (RxC2) соответствующие биты данных RxD1 (RxD2) запоминаются в устройстве DTE3 (DTE4).

По линии передаются сигналы DL1 и DL2 – аналоги сигналов DATA и DELIM 2, показанных на рис. 3.11. Линия может быть выполнена в виде двух витых пар проводов, двух оптоволоконных световодов или иных средств передачи сигналов, в том числе, содержащих ретрансляторы.

Все процессы, протекающие в системе, синхронизированы сигналами от генератора G, размещенного в передатчике (рис. 3.14).

Передатчик работает следующим образом. Сигнал CL1 (TxС1) с выхода генератора G передается в устройство DTE1 (рис. 3.15). В ответ на этот сигнал из устройства DTE1 выдаются данные TxD1 и поступают на вход двухразрядного сдвигового регистра RG.

Под действием положительных фронтов сигнала CL2 данные последовательно продвигаются через этот регистр, выдаются через усилитель в линию и одновременно с этим анализируются элементом ИЛИ (сумматором по модулю два). До тех пор, пока биты в регистре RG чередуются (...010101...), формируется сигнал $V = 1$, который открывает элемент И по нижнему входу. На его выходе формируется синхросигнал TxС2, в ответ на который из устройства DTE2 поступают биты данных TxD2 дополнительного канала передачи. Эти биты проходят через мультиплексор MUX, триггер, усилитель и поступают в линию.

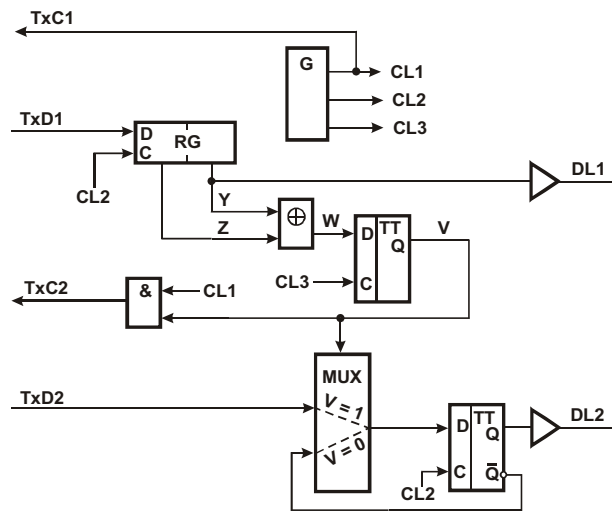


Рис. 3.14. Схема передатчика

Обнаружение элементом Искключающее ИЛИ одинаковых битов в регистре RG означает невозможность передачи очередного “полезного” бита по дополнительному каналу (из-за отсутствия перепада уровней сигнала в основном канале). Поэтому работа с устройством DTE2 приостанавливается сигналом $V = 0$. Этот сигнал закрывает элемент И по нижнему входу и переключает мультиплексор MUX в состояние, при котором нижний (по схеме) триггер работает в счетном режиме, т. е. в каждом такте инвертирует хранящийся в нем бит. Таким образом, сигнал DL2 восполняет “отсутствующие” фронты сигнала DL1 (см. вертикальные стрелки на диаграмме).

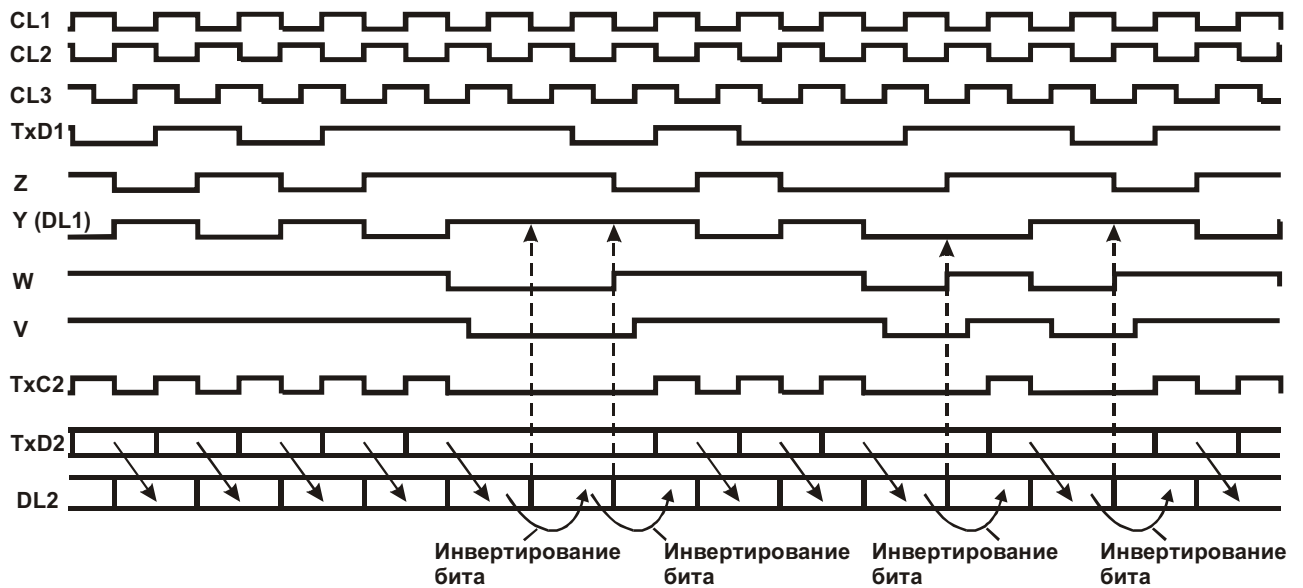


Рис. 3.15. Временные диаграммы сигналов передатчика

Приемник (рис. 3.16, рис. 3.17) работает следующим образом.

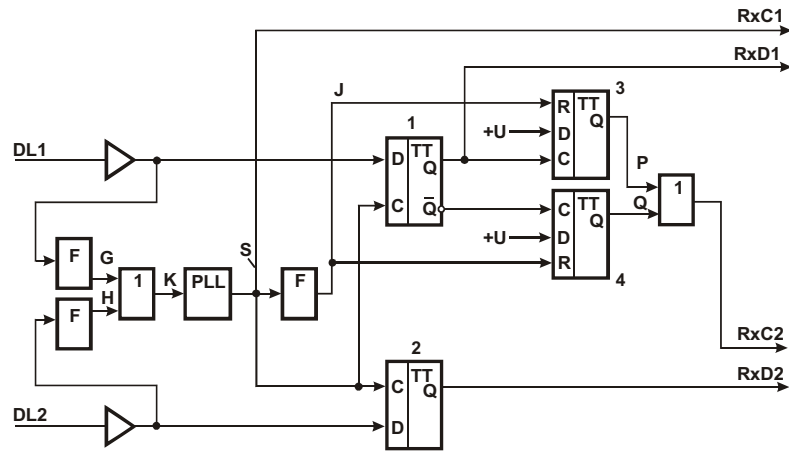


рис. 3.16. Схема приемника

Сигналы DL1 и DL2 проходят через входные усилители и формирователи F импульсов. Сигнал G сопровождает любое изменение сигнала DL1. В данном примере в сигнале G отсутствуют импульсы на позициях, соответствующих вертикальным стрелкам между двумя верхними диаграммами. Сигнал H сопровождает любое изменение сигнала DL2. Так как сигнал DL2 показан на диаграмме условно, без указания конкретного кода, то заранее не известно, будут ли сформированы импульсы, помеченные на диаграмме сигнала H крестиками. Однако можно утверждать, что остальные импульсы на этой диаграмме будут обязательно сформированы, так как они соответствуют гарантированным изменениям сигнала DL2 при его инвертировании. Таким образом, с помощью элемента ИЛИ “недостающие” импульсы восполняются, и сигнал K представляет собой периодическую последовательность импульсов без “пробелов”. Этот сигнал поступает на вход синхронизации блока PLL фазовой автоподстройки частоты.

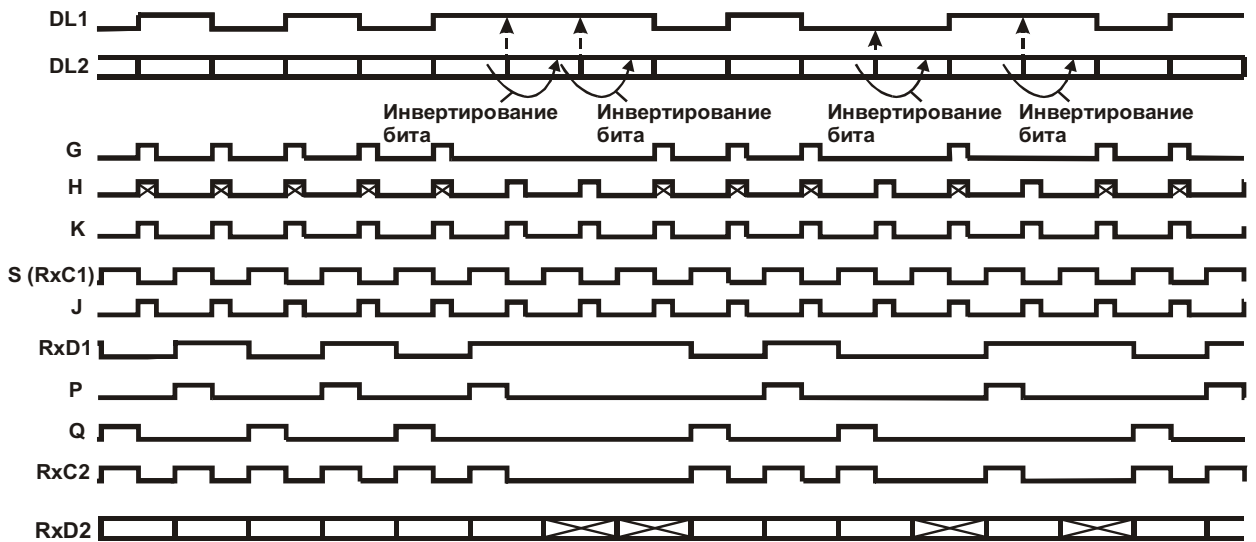


рис. 3.17. Временные диаграммы сигналов приемника

Блок фазовой автоподстройки частоты предназначен для формирования высокостабильного синхросигнала S (RxC1) на основе непрерывного слежения за входным сигналом K. В данном примере отрицательный фронт сигнала S привязан к положительному фронту сигнала K. Благодаря достаточной “инерционности” блока сигнал S практически нечувствителен к дрожанию фазы сигнала K и иным его кратковременным искажениям, вызванным помехами в линии.

По положительному фронту сигнала S принимаемые из линии данные записываются в триггеры 1 и 2. Сигналы с выходов этих триггеров поступают в устройства DTE3 и DTE4. Как отмечалось, сигнал $RxD2$ содержит не только полезные, но и служебные биты, полученные инвертированием предыдущих.

Служебные биты сигнала $RxD2$ (помеченные крестиками на нижней диаграмме), в отличие от полезных, не сопровождаются отрицательными фронтами сигнала $RxC2$ и поэтому не воспринимаются устройством DTE4. Для “прореживания” импульсов $RxC2$ используются триггеры 3, 4 и элемент ИЛИ. Триггер 3 устанавливается в единичное состояние по положительному фронту сигнала $RxD1$, а триггер 4 – по отрицательному фронту этого сигнала. Возврат этих триггеров в нулевое состояние происходит под действием импульсов J , которые поступают с формирователя импульсов и соответствуют отрицательным фронтам сигнала S . Сигналы P и Q с выходов триггеров 3 и 4 суммируются элементом ИЛИ. Поэтому в периоды стабильности сигнала $RxD1$ синхросигнал $RxC2$ не вырабатывается, что и требуется для фильтрации служебных битов.

Идея использования сигнала разграничения битов для передачи кадровых меток

Предлагаемое решение, так же как и предыдущее, расширяет функциональные возможности схемы передачи данных, использующей сигнал разграничения битов (см. рис. 3.10). Точнее, простыми средствами создается некий дополнительный канал связи, по которому передаются признаки, определяющие положение межкадровых границ (рис. 3.18).

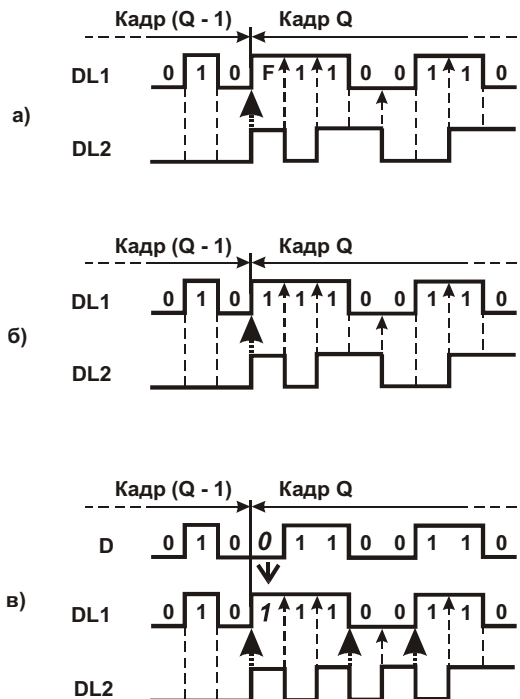


рис. 3.18. Временные диаграммы, поясняющие идею передачи кадровых меток:
a – с применением однобитового флага F ; *б, в* – без применения флага

В предлагаемом решении кадры могут передаваться двумя основными способами. Первый способ (рис. 3.18, *a*) подразумевает наличие флагового (служебного) бита F в начале каждого кадра. Второй способ (рис. 3.18, *б, в*) применим к передаче кадров “стык в стык”, без каких-либо служебных битов вблизи межкадровых границ. Второй способ предпочтителен, так как данные передаются с большей скоростью благодаря исключению из потока лишних (с точки зрения потребителя данных) битов F . Но этот

способ, в отличие от первого, требует некоторых “интеллектуальных затрат” со стороны потребителя данных, так как в среднем в половине ситуаций нужно не только вычислить положение межкадровой границы, но и проинвертировать первый бит кадра, как будет показано далее. Рассмотрим оба способа подробнее.

Первый способ позволяет обозначить межкадровую границу выполнением двух правил.

Правило 1.1. Флаговый бит F кадра Q (см. рис. 3.18, *a*) выбирается противоположным последнему биту предыдущего кадра $Q - 1$. В данном примере эти биты равны соответственно 1 и 0. Это обеспечивает гарантированный перепад уровней сигнала $DL1$ на межкадровой границе.

Правило 1.2. На межкадровой границе сигнал $DL2$ изменяет состояние. В данном примере этот сигнал на межкадровой границе переходит из 0 в 1.

В результате выполнения обоих правил создается уникальная кодовая ситуация, при которой изменению сигнала $DL2$ соответствует изменение (а не статическое состояние) сигнала $DL1$. Эта ситуация обозначена утолщенной стрелкой и может рассматриваться как допустимое нарушение правил кодирования, принятых в прототипе (см. рис. 3.10). Допустимость такого нарушения подтверждается тем, что межкадровая граница задается одновременно изменяющимися сигналами $DL1$ и $DL2$, что не может ухудшить качество синхронизации между приемником и передатчиком.

Второй способ позволяет обозначить межкадровую границу выполнением двух правил.

Правило 2.1. Если по обеим сторонам межкадровой границы присутствуют разноименные биты (01 или 10), то сигнал $DL2$ на этой границе изменяет состояние, (см. рис. 3.18, *b*). Эта ситуация, по существу, аналогична рассмотренной ранее (см. рис. 3.18, *a*) и отличается от нее только тем, что вместо служебного бита F передается полезный бит, который “случайно” оказался противоположным по отношению к предыдущему.

Правило 2.2. Если по обеим сторонам межкадровой границы первоначально присутствуют одноименные биты (00 или 11), то перепад уровней сигнала $DL1$ на этой границе создается искусственно, инвертированием первого бита кадра Q , как показано на рис. 3.18, *в*. В приведенном примере исходные данные D содержат нулевые биты по обеим сторонам межкадровой границы, но, благодаря инвертированию первого бита кадра Q , сигнал данных $DL1$ с искусственно внесенной ошибкой содержит перепад уровней на этой границе. Чтобы отличить эту ситуацию от предыдущей (см. правило 2.1), в сигнал $DL2$ вводится не одно, а ряд описанных ранее нарушений правил кодирования. Первое из них соответствует межкадровой границе. В данном примере используются три следующих подряд нарушения правил, хотя возможны и иные “договоренности” между передатчиком и приемником. Приемник обнаруживает обусловленную “договором” последовательность нарушений правил, по первому из них определяет межкадровую границу и инвертирует первый бит кадра Q , восстанавливая исходную последовательность битов (см. диаграмму сигнала D).

Чтобы защититься от тривиальных кадров, например типа “все нули” или “все единицы”, в которых вообще отсутствуют перепады сигнала данных, применяется скремблирование. В скремблированном потоке данных в среднем каждый второй битовый интервал содержит перепад уровней сигнала.

Пример схемной реализации системы с разграничением кадров

Рассмотрим пример реализации только что описанной идеи. Система передачи данных, показанная на рис. 3.19, предназначена для пересылки кадров из устройства DTE1 в устройство DTE2.



Рис. 3.19. Система передачи данных, использующая дополнительный канал связи для передачи кадровых меток

Пара сигналов $TxC - TxD$ обеспечивает синхронную передачу данных из устройства $DTE1$ в передатчик. Положительные фронты сигнала TxC задают границы битовых интервалов. По отрицательным фронтам сигнала TxC соответствующие биты данных TxD временно запоминаются в передатчике. Признак FR начала кадра формируется устройством $DTE1$ в виде сигнала лог. 1 и передается на вход передатчика одновременно с первым битом этого кадра.

Пара сигналов $RxC - RxD$ на выходах приемника обеспечивает синхронную передачу данных в устройство $DTE2$. Положительные фронты сигнала RxC задают границы битовых интервалов. По отрицательным фронтам сигнала RxC соответствующие биты данных RxD запоминаются в устройстве $DTE2$. Признак $FRAME$ начала кадра формируется приемником в виде сигнала лог. 1 и передается на вход устройства $DTE2$ одновременно с первым битом этого кадра.

Далее рассмотрены три режима работы системы передачи данных. В первом режиме данные передаются без кадровых синхроимпульсов FR ($FRAME$). Во втором режиме передаваемые кадры помечены однобитовыми флагами F . В третьем режиме кадры не содержат флаговых битов.

Первый режим

Все процессы, протекающие при передаче данных, синхронизированы сигналами от генератора G (рис. 3.20). Сигнал $CL1$ (TxC) с выхода генератора передается в устройство $DTE1$. В ответ на этот сигнал из устройства $DTE1$ выдаются данные TxD . Они поступают на вход данных двухразрядного сдвигового регистра RG . Под действием положительных фронтов сигнала $CL2$ на входе синхронизации этого регистра данные сдвигаются в нем, выдаются через усилитель в линию и одновременно с этим анализируются элементом Искключающее ИЛИ 1. До тех пор, пока биты в регистре RG чередуются (...010101...), формируется сигнал $W = 1$, который поддерживает элемент 2 в режиме инвертирования сигнала, поступающего с нулевого выхода триггера 4.

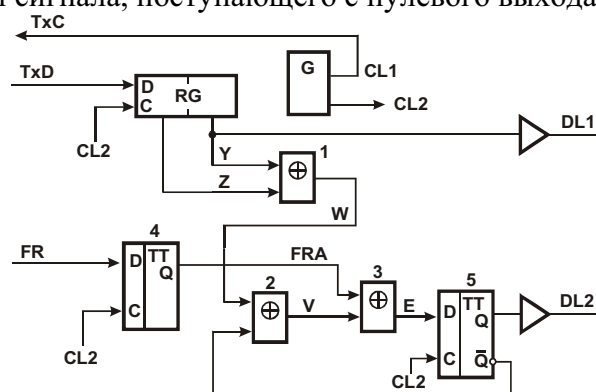


Рис. 3.20. Схема передатчика

Так как в данном режиме сигнал FR кадровой синхронизации отсутствует (постоянно равен лог. 0), триггер 4 постоянно находится в нулевом состоянии ($FRA = 0$). При этом элемент 3 работает в режиме повторения сигнала V ($E = V$). Сигнал с нулевого плеча триггера 5 возвращается на его вход данных после инвертирования элементом 2, поэтому в каждом такте текущее состояние этого триггера подтверждается. Сигнал $DL2$ в линии остается неизменным (0 или 1).

Обнаружение элементом 1 одинаковых битов в регистре RG означает, что необходимо создать перепад уровней сигнала DL2. В этой ситуации сигнал $W = 0$ переводит элемент 2 в режим повторения сигнала с нулевого выхода триггера, поэтому триггер включается в счетный режим, при котором его состояние под действием тактового импульса CL2 изменяется на противоположное.

Сигналы DL1 и DL2 поступают в приемник (рис. 3.21 - рис. 3.23), проходят через усилители и формирователи F импульсов. Сигнал G сопровождает любое изменение сигнала DL1. В данном примере в сигнале G импульсы отсутствуют на позициях, соответствующих вертикальным стрелкам на диаграммах. Сигнал H сопровождает любое изменение сигнала DL2. С помощью элемента ИЛИ “недостающие” импульсы сигнала G восполняются, и сигнал K представляет собой периодическую последовательность импульсов без “пробелов”. Этот сигнал поступает на вход блока PLL фазовой автоподстройки частоты.

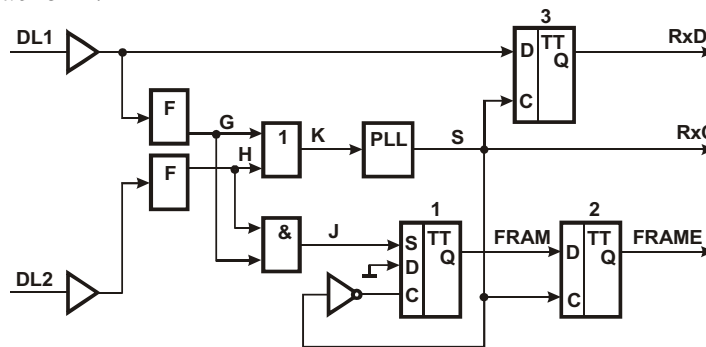


рис. 3.21. Схема приемника

Блок фазовой автоподстройки частоты предназначен для формирования высокостабильного синхросигнала S (RxC) на основе непрерывного “инерционного” слежения за входным сигналом K. Так как в данном режиме импульсы G и H никогда не формируются одновременно, сигнал J на выходе элемента ИЛИ постоянно равен 0, триггеры 1 и 2 находятся в нулевом состоянии. Принятые из канала связи данные проходят через триггер 3 и передаются в устройство DTE2.

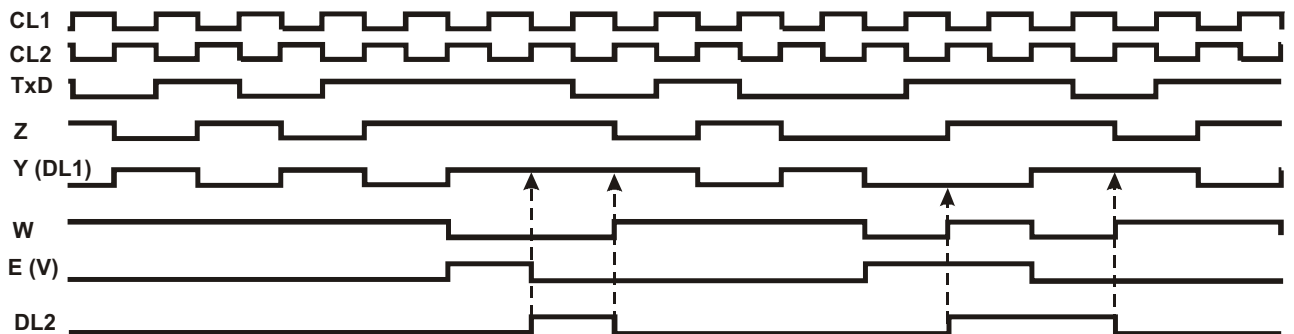


рис. 3.22. Временные диаграммы сигналов передатчика в отсутствие кадровых синхроимпульсов

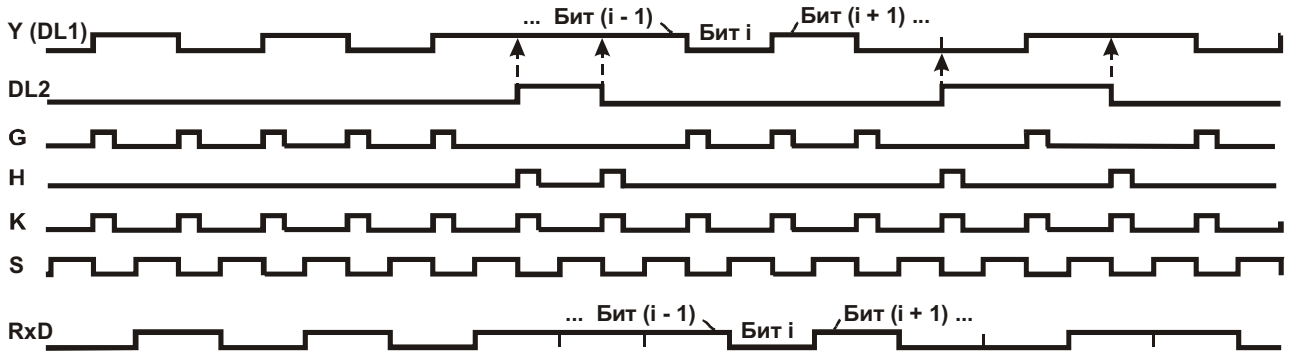


рис. 3.23. Временные диаграммы сигналов приемника в отсутствие кадровых синхросигналов

Второй режим

Этот режим отличается от предыдущего тем, что из устройства DTE1 поступают импульсы FR, соответствующие флаговому биту передаваемых кадров, (рис. 3.24, рис. 3.25). Это приводит к одновременному формированию фронтов сигналов DL1 и DL2. В момент обнаружения межкадровой границы формируется сигнал $J = 1$, который устанавливает триггер 1 (см. рис. 3.21) в единичное состояние. Сигнал с выхода этого триггера проходит через триггер 2 и поступает в устройство DTE2.

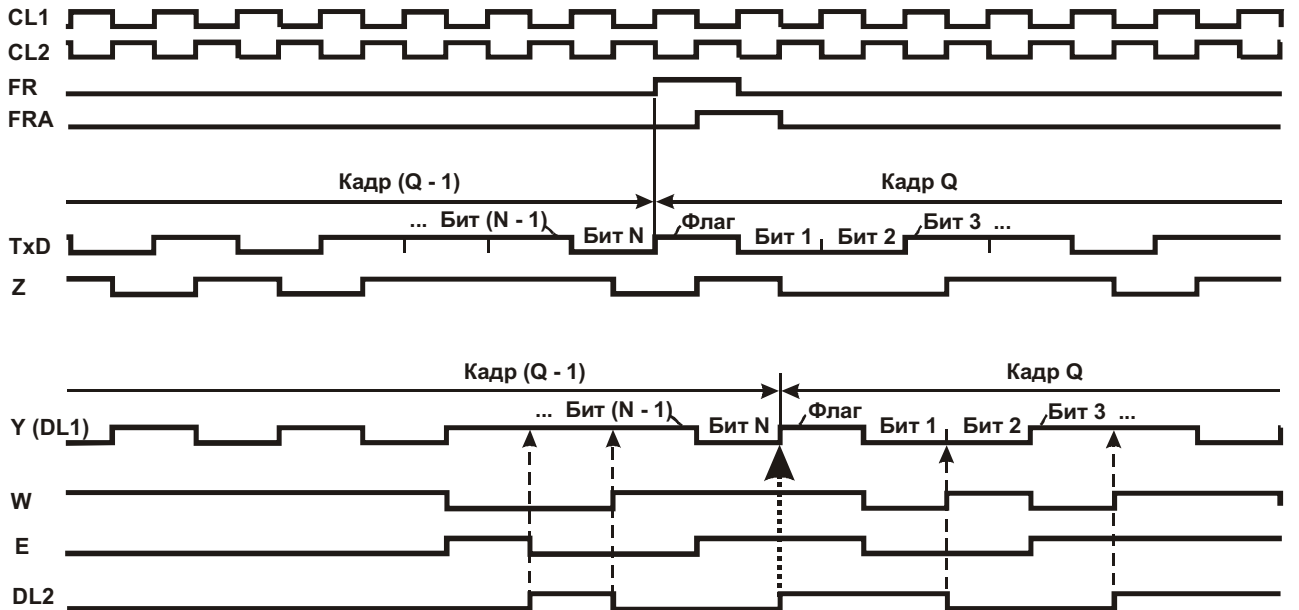


рис. 3.24. Временные диаграммы сигналов передатчика при пересылке кадров с однобитовыми флагами

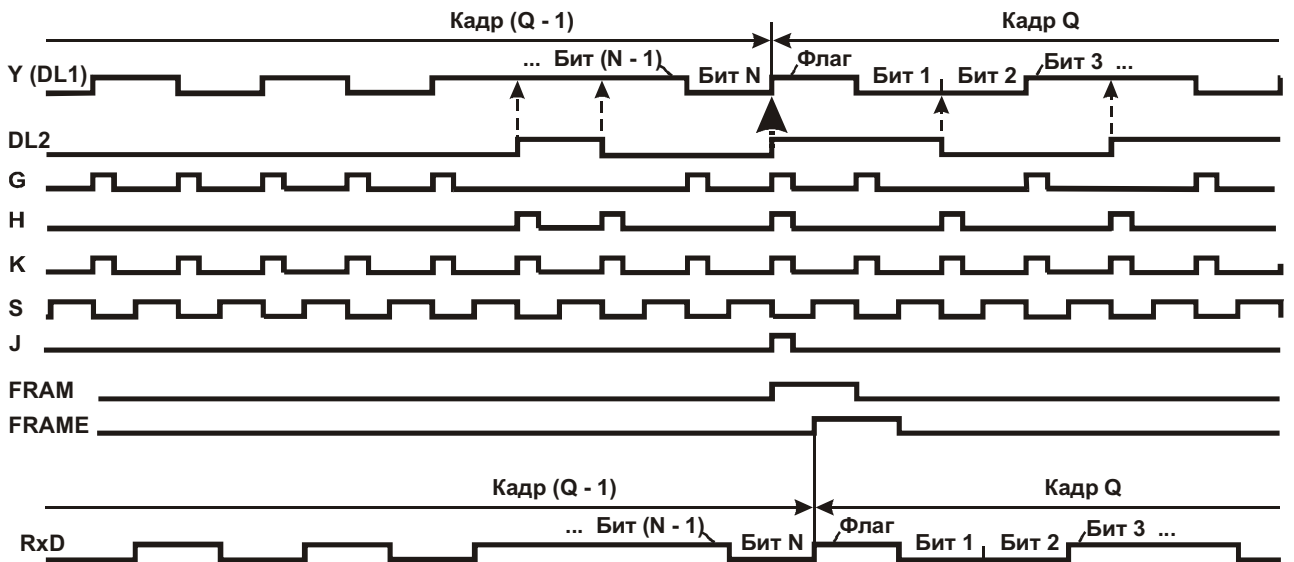


рис. 3.25. Временные диаграммы сигналов приемника при пересылке кадров с однобитовыми флагами

Третий режим

В этом режиме возможны две ситуации. Первая характеризуется наличием противоположных битов по обеим сторонам межкадровой границы. В этом случае временные диаграммы, приведенные на рис. 3.26 и рис. 3.27, отличаются от диаграмм на рис. 3.24 и рис. 3.25 только тем, что вместо флагового бита передается первый "полезный" бит кадра.

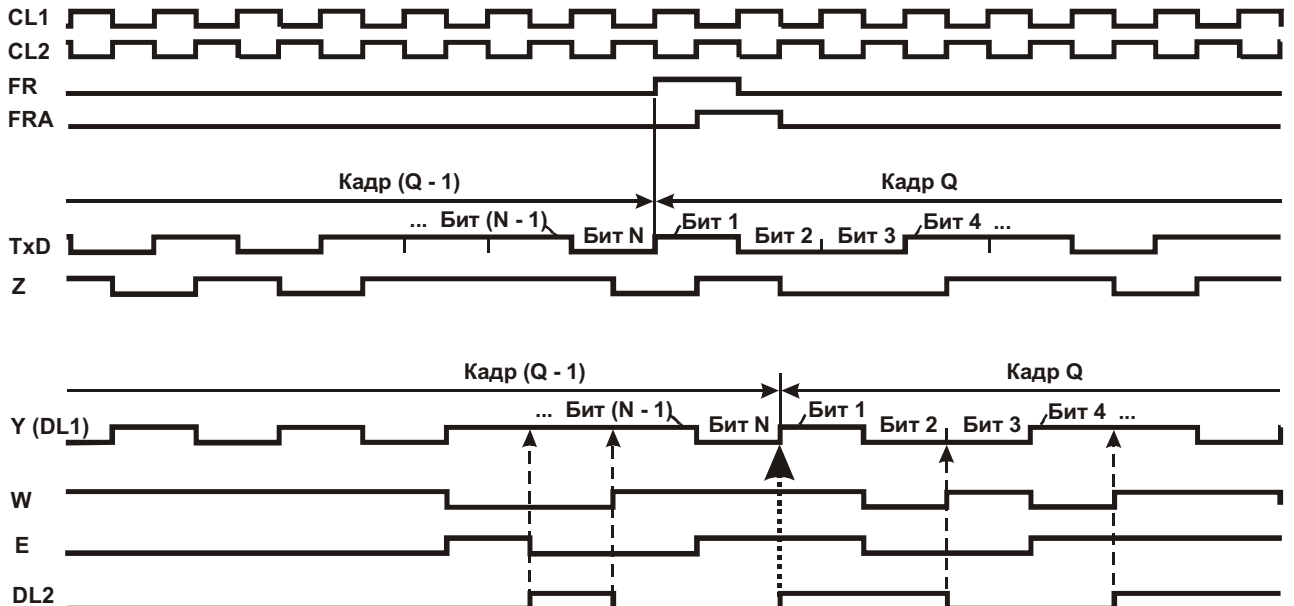


рис. 3.26. Временные диаграммы сигналов передатчика при пересылке бесфлаговых кадров с разноименными битами по обеим сторонам межкадровой границы

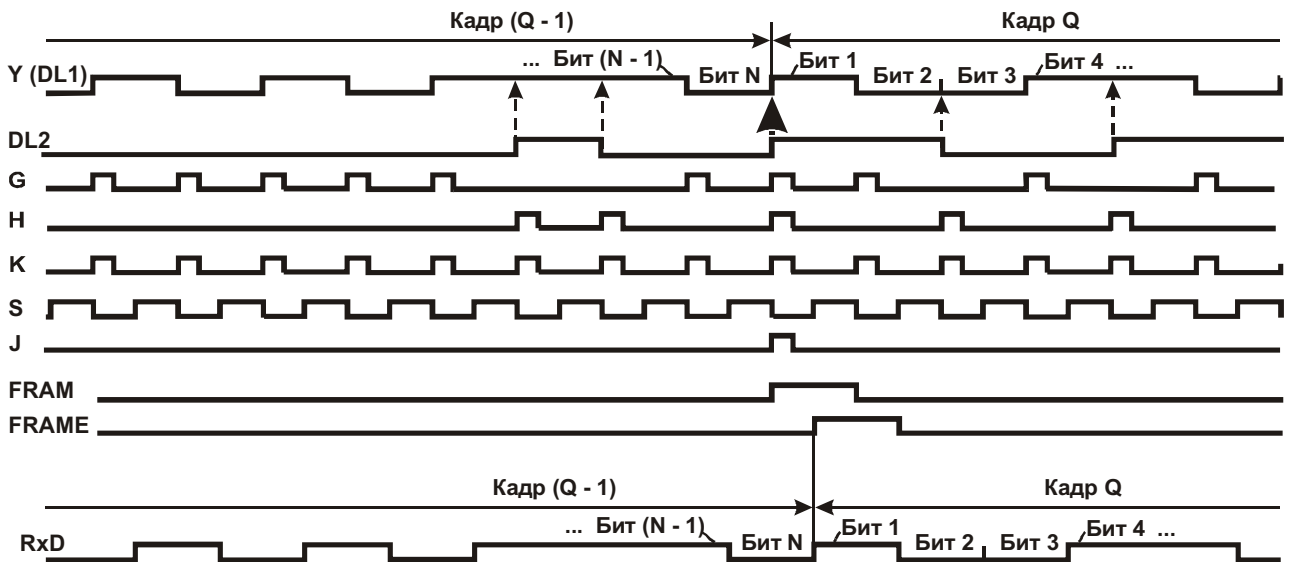


Рис. 3.27. Временные диаграммы сигналов приемника при пересылке бесфлаговых кадров с разноименными битами по обеим сторонам межкадровой границы

Вторая ситуация (рис. 3.28, рис. 3.29) также характеризуется наличием противоположных битов по обеим сторонам межкадровой границы, но, как уже отмечалось, первый бит кадра преднамеренно искажен для создания гарантированного перепада уровней сигнала DL1 на межкадровой границе. Чтобы отличить вторую ситуацию от первой, вместо одного передаются три кадровых импульса.

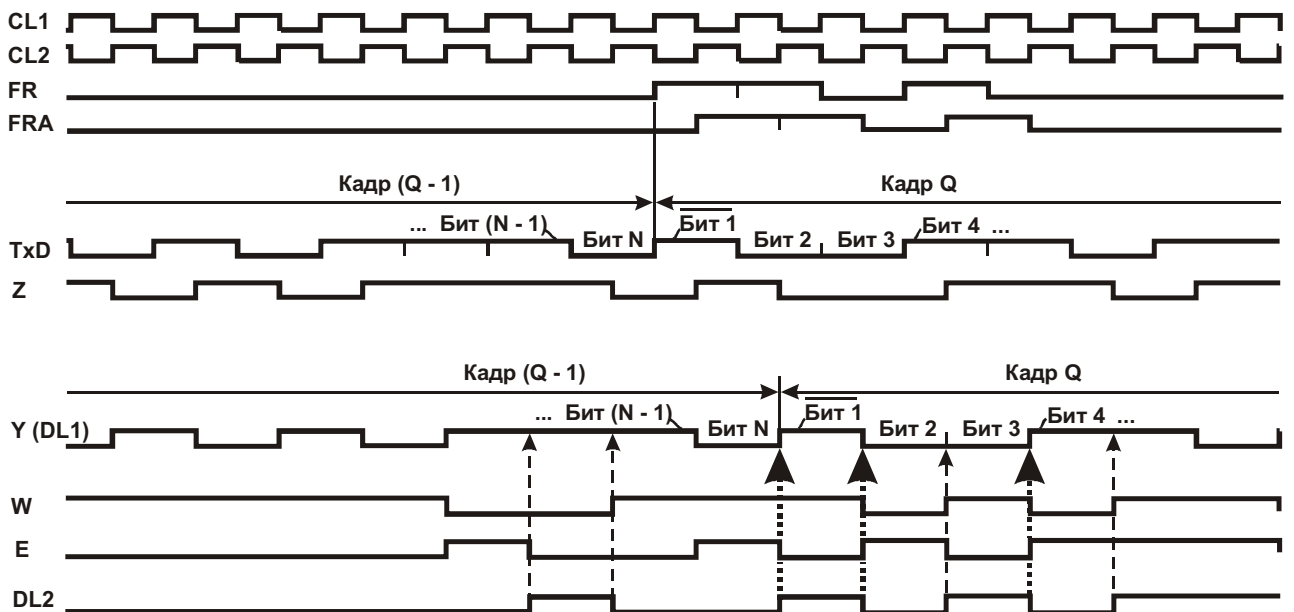


Рис. 3.28. Временные диаграммы сигналов передатчика при пересылке бесфлаговых кадров с одноименными битами по обеим сторонам межкадровой границы

Из диаграмм, приведенных на рис. 3.28, видно, что из устройства DTE1 периодически поступают триады кадровых импульсов. Это приводит к появлению трех событий одновременного формирования фронтов сигналов DL1 и DL2. Сигнал J = 1 формируется в приемнике троекратно и воздействует на вход установки триггера 1 в единицу. Отметим, что в D-триггере сигнал S установки единицы имеет более высокий приоритет, чем сигнал на динамическом входе С. Сигнал с выхода триггера 1 проходит

через триггер 2 и поступает на вход устройства DTE2. Как уже отмечалось, получение устройством DTE2 серии близлежащих кадровых импульсов свидетельствует о необходимости инвертирования первого бита кадра.

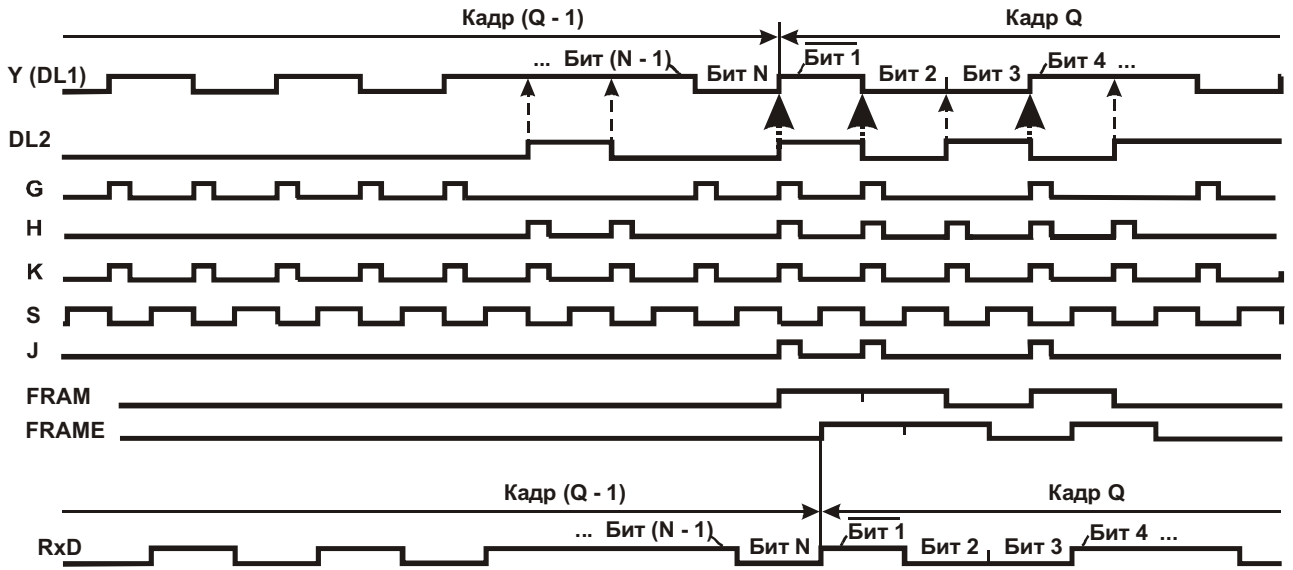


Рис. 3.29. Временные диаграммы сигналов приемника при пересылке бесфлаговых кадров с одноименными битами по обеим сторонам межкадровой границы

Возможны также иные способы передачи данных с помощью предлагаемого устройства. Например, для повышения помехоустойчивости можно оперировать не одним и тремя кадровыми импульсами, а группами из M и $2M$ импульсов, где $M > 10$. Канал кадровых импульсов можно одновременно использовать для передачи дополнительного потока данных и т. п.

Выводы.

Рассмотренные в п. 3.3 решения позволяют сделать следующие выводы.

1. Классический способ передачи данных между устройствами (см. рис. 3.6), при котором истинность битов TxD подтверждается отрицательными фронтами синхросигнала CLK, не обеспечивает высокой скорости обмена. Рассмотрены два направления улучшения классического способа передачи данных.

2. Первое направление основано на использовании как отрицательных, так и положительных фронтов синхросигнала CLK для подтверждения истинности битов данных TxD (см. рис. 3.7). Скорость передачи данных при неизменной частоте синхросигнала CLK повышается в два раза.

3. Второе направление подразумевает изменение функций синхросигнала. Этот сигнал (именуемый сигналом разграничения DELIM) формируется таким образом, что его форма зависит от передаваемых данных (см. рис. 3.10). Точнее, сигнал остается неизменным, если изменяются данные, и, наоборот, сигнал изменяется, если данные постоянны. Такое решение, так же как и предыдущее (см. рис. 3.7), повышает скорость передачи данных в два раза и, кроме того, разравнивает спектр синхросигнала (DELIM), что способствует снижению уровня излучаемых в линию помех.

4. Второе направление не исчерпывается только что упомянутым способом формирования сигнала DELIM. При синтезе этого сигнала открываются достаточно широкие возможности замены его безразличных состояний битами данных. Основное следствие этого состоит в том, что сигнал DELIM может переносить данные параллельно с их переносом по линии данных (см. рис. 3.11). При этом с увеличением числа линий данных эффективность такого решения возрастает (см. рис. 3.12).

5. Способность переноса данных сигналом DELIM наряду с их переносом по штатной линии данных использована в двух примерах построения телекоммуникационных систем. Первый пример иллюстрирует возможность построения двухканальной системы (см. рис. 3.13), второй – возможность синхронизации приемника с передатчиком на уровне информационных кадров, причём флаговые биты, соответствующие началу кадров, могут отсутствовать (см. рис. 3.19).

3.4. Аппаратное управление потоком данных с использованием пачек сигналов TxC

Рассмотренные далее решения (п. 3.4.1, 3.4.2) объединяет способ управления потоком данных TxD между устройствами DTE и DCE. Этот способ заключается в том, что вместо непрерывного синхросигнала использованы пачки сигналов TxC. В зависимости от интенсивности формирования импульсов TxC изменяется интенсивность передачи битов данных TxD. Это позволяет упростить схемы согласования скоростей передачи данных при работе с линией, исключив линии RTS и CTS, а также улучшить структуру системы синхронизации формирователя информационных кадров.

3.4.1. Согласование средней скорости передачи данных между устройствами DTE и DCE со скоростью передачи данных по линии

В отличие от традиционного решения задачи аппаратного управления потоком данных (hardware flow control, см. п. 1.2) сигналами RTS и CTS, в [6] предлагается возложить функцию регулирования скорости одного из потоков на синхросигнал TxC.

В примере построения схемы, реализующей эту идею (рис. 3.30), компьютер (устройство типа DTE, см. гл. 1) обменивается данными с удаленным абонентом (на рисунке не показан) через модем (устройство типа DCE). Компьютер взаимодействует с модемом по интерфейсу RS-232 или иному, принадлежащему этому же семейству. Скорость передачи данных по линии в каждом направлении равна 160 кбит/с. Скорость приема данных RxD в компьютер также равна 160 кбит/с.

Скорость передачи данных TxD из компьютера в модем выбирается большей, чем скорость передачи данных по линии, и в нашем примере составляет 230 кбит/с. Чтобы этот факт не вызывал недоразумений, сразу отметим, что данные TxD передаются из компьютера в модем не непрерывно, а пачками, по мере необходимости пополнения буферной памяти модема. Так что средняя скорость передачи данных из компьютера в модем равна скорости их выдачи в линию и составляет 160 кбит/с, в отличие от мгновенной скорости (230 кбит/с), на которой передаются данные в пачке.

Напомним, что синхросигнал RxS сопровождает сигнал данных RxD. По получении сигнала TxC компьютер формирует очередной бит TxD, который поступает в модем и принимается им. Соответствующая временная диаграмма рассматривалась ранее (см. рис. 1.6, б).

Модем содержит передатчик, приемник и буферную память с блоком управления (другие компоненты сейчас не представляют для нас интереса). Передатчик непрерывно считывает биты из буферной памяти, смешивает их с синхросигналом C и передает полученную смесь в линию. Приемник выделяет из поступающего от удаленного абонента линейного сигнала данные RxD и синхросигнал C, который поступает на вход RxS компьютера. Этот же синхросигнал C, как отмечалось, управляет считыванием данных из буферной памяти и их выдачей в линию.

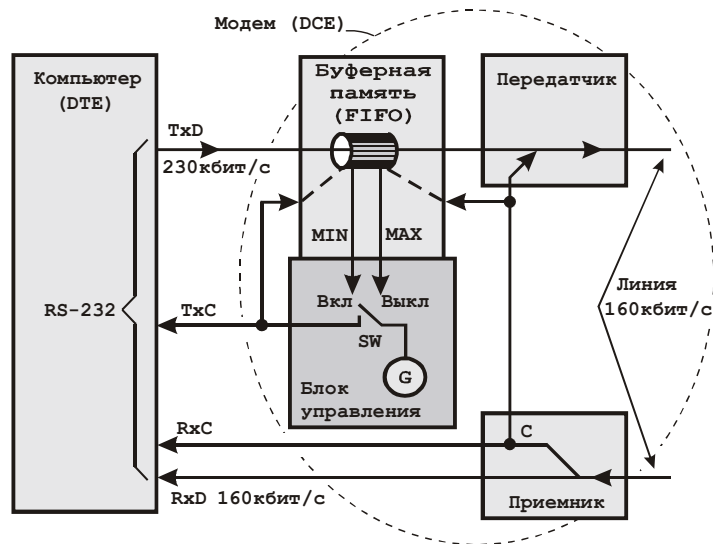


рис. 3.30. Пример схемы с аппаратным управлением потоком данных TxD

Буферная память условно показана в виде трубопровода. Она работает по принципу “первым пришел, первым обслужен” (First Input – First Output или FIFO) [72]*. С такой памятью могут бесконфликтно работать местные источник и приемник данных. Конфликты, которые могли бы возникнуть при одновременных обращениях к памяти по записи и чтению, устраняются аппаратно. Буферная память содержит индикаторы уровня заполнения. Если уровень велик и превышает некоторый порог, например 90%, то срабатывает индикатор максимума MAX, ключ (логический элемент) SW в блоке управления размыкается. Если уровень мал, например менее 10%, то срабатывает индикатор минимума MIN, ключ SW в блоке управления замыкается. Если уровень лежит в пределах от MIN до MAX, то предыдущее состояние ключа сохраняется. Генератор G вырабатывает непрерывную последовательность синхроимпульсов частотой 230 кГц.

В исходном состоянии компьютер непрерывно принимает поступающие из линии данные RxD под управлением сигнала RxС. В то же время под управлением сигнала TxС компьютер передает данные в модем. При этом уровень заполнения буферной памяти равномерно пульсирует в диапазоне от MIN до MAX.

Действительно, предположим, что ключ SW замкнут. Тогда поток данных TxD поступает в буферную память со скоростью 230 кбит/с. Уровень заполнения буфера неуклонно растет, так как скорость рассасывания данных меньше скорости их поступления. В некоторый момент уровень достигает критической отметки (MAX), ключ SW размыкается, компьютер прекращает выдачу данных TxD, но процесс их рассасывания продолжается. Уровень заполнения монотонно снижается, и когда он пересекает границу MIN, ключ SW замыкается и т. д.

Приведенный пример не следует рассматривать только как конкретную схему управления потоком данных – это всего лишь наиболее простая иллюстрация идеи сокращения числа управляющих линий и повышения эффективности использования процессорного времени компьютера и модема. Последнее достигается за счет концентрации данных в компактные пакеты, разделенные интервалами. Эти интервалы можно использовать для тех или иных вычислений.

* В дальнейшем использованы также две альтернативные модели такой памяти – бак с жидкостью и кольцо из ячеек памяти (см., например, п. 4.2). Модель в виде кольца из ячеек памяти позволяет более детально проследить за процессами, протекающими в буферной памяти, но обладает меньшей наглядностью по сравнению с двумя другими.

3.4.2. Упрощение системы синхронизации формирователя HDSL-кадров

Рассмотрим процесс формирования одной из разновидностей так называемых HDSL-кадров*, уделяя основное внимание вопросам синхронизации взаимодействующих устройств.

В системе, показанной на рис. 3.31, данные передаются из устройства 1 в устройство 2 через модемы 1 и 2 в синхронном режиме. Эти модемы наделены способностью формировать кадры при передаче данных в линию и расформировывать кадры, принимаемые из линии. Здесь и далее показана односторонняя передача данных, но, предполагая, что схема симметрична, будем иметь в виду, что все описанные процессы протекают и при транспортировании данных в обратном направлении.

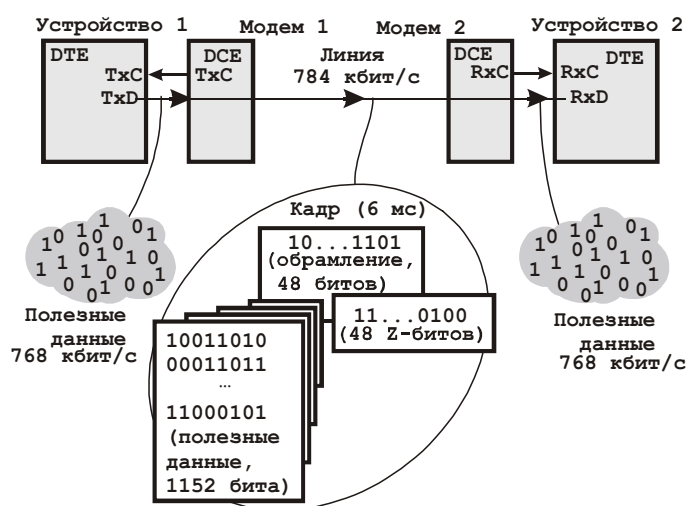


рис. 3.31. Система передачи данных между устройствами 1 и 2

Под управлением синхросигналов TxС устройство 1 передает в модем 1 поток данных со скоростью 768 кбит/с. Модем 1 упаковывает данные в кадры определенного формата. Номинальная длительность кадра – 6 мс (возможны небольшие отклонения в сторону уменьшения или увеличения, но ими пренебрегаем). Как условно показано на рисунке, каждый кадр содержит четыре поля данных по 1152 бита в каждом, а также два служебных поля.

Первое служебное поле определяет некое обрамление кадра – стартовый флаг, контрольную сумму и иные признаки. Его размер – 48 битов. Второе служебное поле такого же размера (48 так называемых Z-битов) представляет собой группу вакантных мест для передачи сообщений между устройствами, способными анализировать структуру кадра, и, в частности, выделять из него Z-биты. Такими устройствами в нашем примере являются модемы 1 и 2. С помощью Z-битов создается невидимый канал связи между модемами, способный передавать информацию с темпом 48 бит за 6 мс или, что то же самое, 8 кбит/с. (Первые восемь из 48 Z-битов зарезервированы “на будущее” – они пока не используются и постоянно установлены в состояние лог. 1; но в данный момент это нас не интересует.)

Такой же вклад (8 кбит/с) в общий поток данных по линии вносит и первое служебное поле. Действительно, в течение каждого 6 мс передаются 48 битов этого поля, а это соответствует тому, что за одну секунду передаются 8 кбит служебных данных. В результате суммарная скорость передачи данных по линии складывается из скоростей трех потоков (полезного и двух служебных) и составляет $768 + 8 + 8 = 784$ кбит/с.

* HDSL – High-bit-rate Digital Subscriber Line – технология высокоскоростной передачи данных по кабелям на основе витых пар медных проводов.

Иными словами, к первоначальному потоку полезных данных из устройства 1 (768 кбит/с) добавляется бесполезный (с точки зрения гипотетического потребителя данных) поток сравнительно малой интенсивности – 16 кбит/с. Накладные расходы составляют $16/768$ или около 2,1%.

Более строгое отображение структуры кадра приведено на рис. 3.32. Служебные биты помечены затененными областями. Из рисунка следует, что поле обрамления из 48 бит представлено разрозненными фрагментами ($16 + 10 + 10 + 10 + 2 = 48$). То же относится и к Z-битам – они перемежаются с 96-битовыми блоками данных. Граница окончания кадра может перемещаться влево или вправо на два бита (см. обозначения “-” и “+”) при соответствующем изменении длины первого служебного поля. В наших расчетах принимаем номинальную длительность кадра – 6 мс.

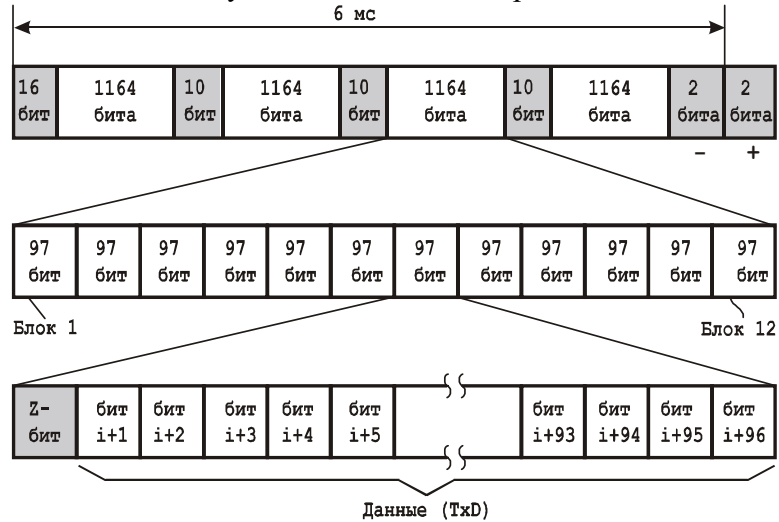


рис. 3.32. Структура одного из вариантов HDSL-кадра.

Проблема заключается в том, что модемы 1 и 2 (см. рис. 3.31) оперируют двумя частотами: 768 и 784 кбит/с. Их можно получить от двух генераторов или от общего синтезатора частот. Однако такие решения слишком громоздки (и это, как будет показано далее, не единственный недостаток). В [7] предлагается в явном виде не использовать частоту 768 кГц и оперировать только частотой 784 кГц, сохраняя структуру потоков в соответствии с рис. 3.31! (При этом “недостающая” частота в неявном виде формируется без дополнительных затрат аппаратуры.)

Идея поясняется схемой и временной диаграммой (рис. 3.33).

В данном примере компьютер и модем выполнены соответственно в виде устройств типа DTE и DCE (см. гл. 1). Компьютер взаимодействует с модемом по одному из интерфейсов семейства RS-232. Передача данных из компьютера в модем выполняется в синхронном режиме, в соответствии с временной диаграммой, которая была показана на рис. 1.6, б.

Скорость передачи данных TxD равна 784 кбит/с при условии, что электронный ключ SW замкнут управляющим сигналом $L = 1$. Если $L = 0$, то скорость передачи равна нулю. При этом, как мы сможем убедиться, средняя скорость равна 768 кбит/с, что и требуется!

Передаваемые из компьютера данные временно накапливаются в буферной памяти типа FIFO. Формирователь кадров добавляет к данным необходимую служебную информацию. Под действием синхросигналов F сформированные кадры непрерывно и равномерно выдвигаются с выхода памяти на вход передатчика. Передатчик кодирует данные и выдает их в линию.

Формирователь кадров выполнен на основе высокоскоростного микропроцессора и обладает достаточно высоким интеллектом. Программа сборки кадра, в частности, знает, что после размещения на входе буферной памяти очередного Z-бита нужно принять

на вход серию из 96 битов данных TxD (см. нижнюю строку диаграммы, представленной ранее на рис. 3.32). Чтобы это осуществить, формирователь кадров выдает сигнал загрузки L = 1, ключ SW замыкается, начинается передача данных TxD.

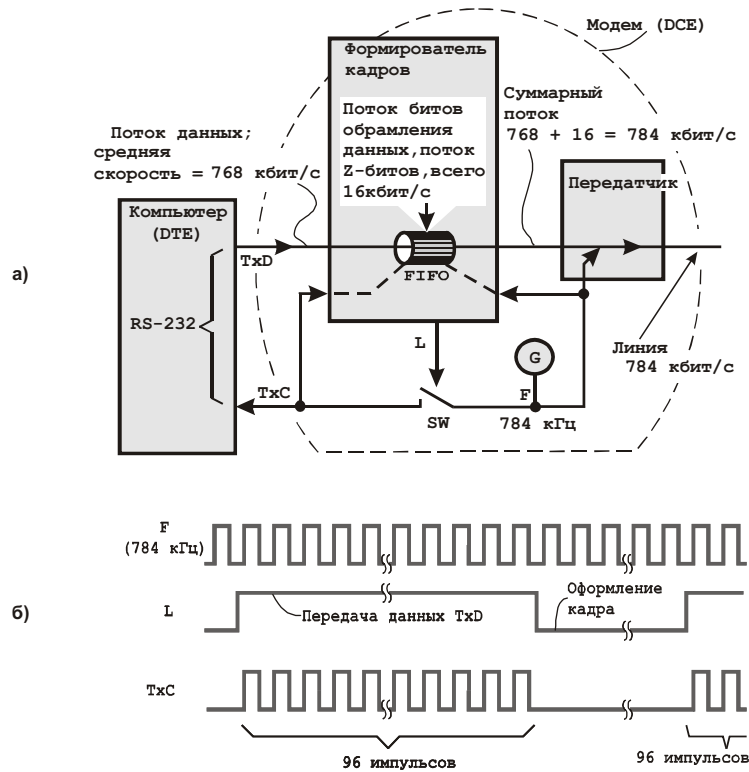


рис. 3.33. Передача данных из компьютера в линию через модем: а – схема синхронизации; б – временные диаграммы

Формирователь отсчитывает 96 импульсов и размыкает ключ SW сигналом $L = 0$. Поток входных данных прекращается. После этого в буферную память вводятся нужные служебные биты, и когда возникает потребность в приеме очередной группы из 96 битов данных, ключ SW вновь замыкается и т. д.

Средняя скорость передачи данных TxD составляет 768 кбит/с. Действительно, с учетом структуры кадра, приведенной на рис. 3.32, за время 6 мс ключ SW замыкается 48 раз, и при каждом замыкании передаются 96 битов данных, всего пересылаются $48 \times 96 = 4608$ битов. Этот темп соответствует передаче $(4608 \times 1000) / 6 = 768\,000$ битов за 1 с, что и требовалось доказать.

Мы уже отмечали одно из преимуществ такого решения – использование только одной частоты синхронизации (другая является как бы виртуальной). Второе преимущество связано с тем, что в процессе оформления кадра нет поступления внешних данных. Это позволяет уменьшить требуемый объем буферной памяти. Кроме того, не нужен аппаратный арбитр, так как принципиально исключены конфликты, которые при иных решениях могли бы возникать при одновременном обращении к буферной памяти со стороны двух источников информации.

Подобное построение системы синхронизации применимо и при расформировании кадра модемом 2 (см. рис. 3.31). В этом случае частота 784 кГц выделяется из линейного сигнала. Данные RxD передаются в устройство 2 аритмично, пачками, со средней скоростью 768 кбит/с.

3.5. Взаимодействие удаленных устройств с использованием дистанционной цифровой коррекции фазы сигнала от синхрогенератора

Дистанционная коррекция фазы сигнала от удаленного синхрогенератора позволяет проектировать системы передачи данных, отвечающие некоторым специфическим требованиям. Одно из таких требований заключается в обеспечении совпадения синхросеток передаваемых и принимаемых данных (п. 3.5.1). Другое требование возникает при проектировании удлинителя сигналов TxС и RxС (в общем случае – несинхронных и несинфазных), когда их необходимо передать на расстояние порядка нескольких километров. Это требование состоит в том, что сигнал TxС должен распространяться “против течения” потока данных (п. 3.5.2).

3.5.1. Выравнивание синхросеток передаваемых и принимаемых данных

В одном из рассмотренных ранее вариантов взаимодействия устройств типа DTE и DCE (см. рис. 2.4) решалась задача получения синхронных и синфазных потоков данных на стыке между устройствами DCE 1 и DCE 2. (Напомним, что эта задача – одна из часто встречающихся на практике при построении синхронных каналов связи между удаленными устройствами.) На рис. 3.34, а представлено решение той же задачи с использованием идеи, предложенной в [8].

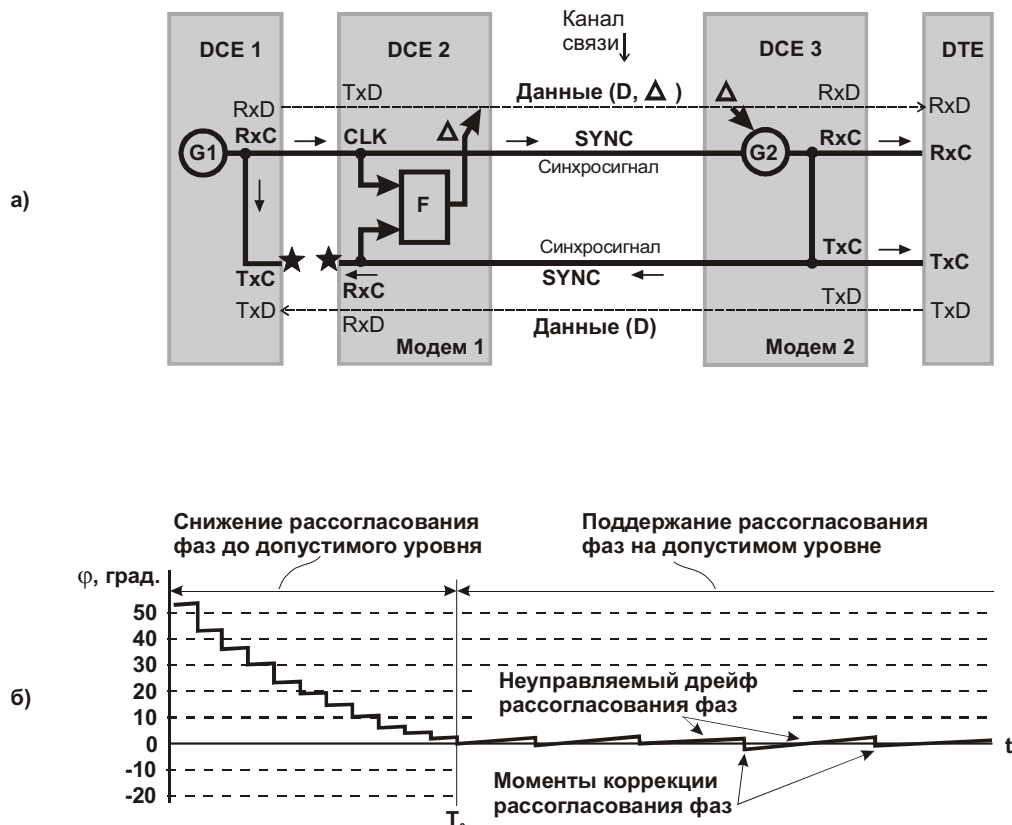


Рис. 3.34. Вариант системы синхронизации двусторонней передачи данных между устройствами DCE 1 и DTE: а – схема; б – временная диаграмма

Рассмотрим задачу более подробно. На практике часто встречаются устройства типа DCE, которые генерируют совпадающие синхросигналы RxС и TxС. В нашем примере устройство DCE 1 (мультиплексор) содержит внутренний генератор G1. Сиг-

нал RxC с выхода генератора $G1$ сопровождает данные, передаваемые в устройство DCE 2, а сигнал TxC с этого же выхода синхронизирует выдачу данных из устройства DCE 2. Таким образом, мультиплексор DCE 1 требует, чтобы его входной поток данных был размещен в той же синхросетке, что и выходной.

Иными словами, сигналы, помеченные на рис. 3.34 звездочками, должны быть синхронными и синфазными (т. е. должны иметь одинаковые частоты и практически совпадающие во времени фронты), несмотря на то, что один из них, распространяясь по каналу связи туда и обратно, возможно, преодолевает до места встречи со своим прототипом многокилометровую петлю и явно опаздывает. Это означает, что сигнал, прошедший длинный путь, должен быть задержан на целое число периодов по отношению к своему прототипу.

Сигнал RxC , помеченный звездочкой, сопровождает данные, передаваемые из устройства DCE 2 в устройство DCE 1. Этот сигнал в явном виде не используется, но его совпадение с сигналом TxC (также помеченным звездочкой) гарантирует правильную передачу данных в устройство DCE 1.

Сигнал RxC с выхода генератора $G1$ поступает на вход CLK устройства DCE 2 (модем 1). Модем 1 временно накапливает данные, поступающие от устройства DCE 1, формирует из них кадры и после кодирования передает их в канал связи. Сигнал в канале связи представляет собой смесь данных D с синхроимпульсами SYNC. Помимо полезных данных в кадре передаются служебные, такие как заголовок, контрольная сумма и т. д. Среди служебных данных содержится некоторый код Δ . Число разрядов этого кода может быть равно, например, трем.

Модем 2 принимает сигнал из канала связи, выделяет из него синхроимпульсы и данные, отделяет полезные данные от служебных и передает полезные данные в устройство DTE. Истинность этих данных подтверждается синхросигналом RxC , который поступает на одноименный вход устройства DTE. В свою очередь, устройство DTE в такт с запросным сигналом TxC выдает данные в модем 2. Эти данные после соответствующих преобразований передаются по каналу связи в модем 1 и далее в устройство DCE 1.

Выделение синхроимпульсов из канального сигнала модемом 2 осуществляется с помощью ведомого генератора $G2$, выполненного по схеме с фазовой автоподстройкой частоты. Особенность этого генератора в том, что под действием управляющего кода Δ он может вносить фазовое смещение выходного сигнала в ту или иную сторону на большую или меньшую величину. Например, при $\Delta = 000_2$ генератор $G2$ не изменяет имеющийся фазовый сдвиг выходного сигнала; при $\Delta = 001_2$ или $\Delta = 010_2$ выходной сигнал дополнительно смещается вправо или влево относительно предыдущего положения на один градус и т. д.

Фазовый компаратор F сравнивает фазы сигналов на его входах (фактически это сигналы, помеченные звездочками). Первый сигнал поступает от генератора $G1$ по короткому пути, второй формируется из канального сигнала с помощью обычного генератора с фазовой автоподстройкой частоты (этот генератор на рисунке не показан). Если фронты сигналов практически совпадают (а это и есть условие правильной работы системы), то $\Delta = 000_2$, генератор $G2$ сохраняет “статус-кво”.

При обнаружении заметного и устойчивого фазового расхождения однократно формируется ненулевой код Δ , который вводится в ближайший кадр и пересылается в модем 2. Генератор $G2$ реагирует на получение этого кода так, что фазовое расхождение сигналов на входах компаратора уменьшается. Если это расхождение снизилось до допустимых пределов, то состояние системы не изменяется; в противном случае проводится новый сеанс коррекции фазы и т. д. Диапазон регулирования фазы сигнала RxC относительно сигнала TxC (тот и другой помечены звездочками) равен ± 180 град.

На рис. 3.34, б показана временная диаграмма регулирования рассогласования фаз ϕ сигналов RxС и TxС. В исходном состоянии (до момента T_0) рассогласование недопустимо велико, поэтому устройство DCE 1 принимает от устройства DCE 2 неправильные данные. Возможность уменьшения рассогласования фаз сохраняется благодаря тому, что канал связи способен передавать данные от устройства DCE 1 к устройству DTE. После ряда шагов коррекции рассогласования фаз (к моменту T_0) передаваемые и принимаемые данные на стыке между устройствами DCE 1 и DCE 2 оказываются размещенными практически в одной и той же синхросетке, что и требовалось. Устройство DCE 1 начинает принимать правильные данные. В дальнейшем (после момента T_0) рассогласование фаз поддерживается в допустимых пределах, как уже было описано.

3.5.2. Передача синхросигнала “против течения” потока данных

Идею коррекции фазы сигнала от удаленного синхрогенератора, описанную в п. 3.5.1, можно применить к решению задачи параллельного распространения по каналу связи двух синхросигналов, один из которых как бы движется против течения. Эта задача уже рассматривалась (см. п. 2.3, рис. 2.5). Для удобства изложения в этот рисунок внесены некоторые изменения и дополнения (рис. 3.35).

Напомним содержание задачи. Предположим, что в нашем распоряжении имеются два устройства: мультиплексор DCE 1 и компьютер DTE 2 (рис. 3.35, а). (Нумерация устройств “привязана” к последующим рисункам.) На выходе мультиплексора можно наблюдать два синхросигнала от двух в общем случае независимых генераторов G_1 и G_2 . Сигнал RxС сопровождает передаваемые данные RxD; сигнал TxС служит запросом на получение данных TxD.

Если мультиплексор и компьютер расположены рядом, например, на расстоянии нескольких метров друг от друга, то их можно соединить кабелем по классической схеме, согласно которой объединяются одноименные контакты соединителей обоих устройств (см. гл. 1). Если же расстояние между мультиплексором и компьютером велико (например, составляет 5 км), то уместно применить для передачи данных удлинитель в виде пары модемов, как показано на рис. 3.35, б. На первый взгляд, все верно, но при более внимательном рассмотрении этой схемы возникают сомнения в ее работоспособности.

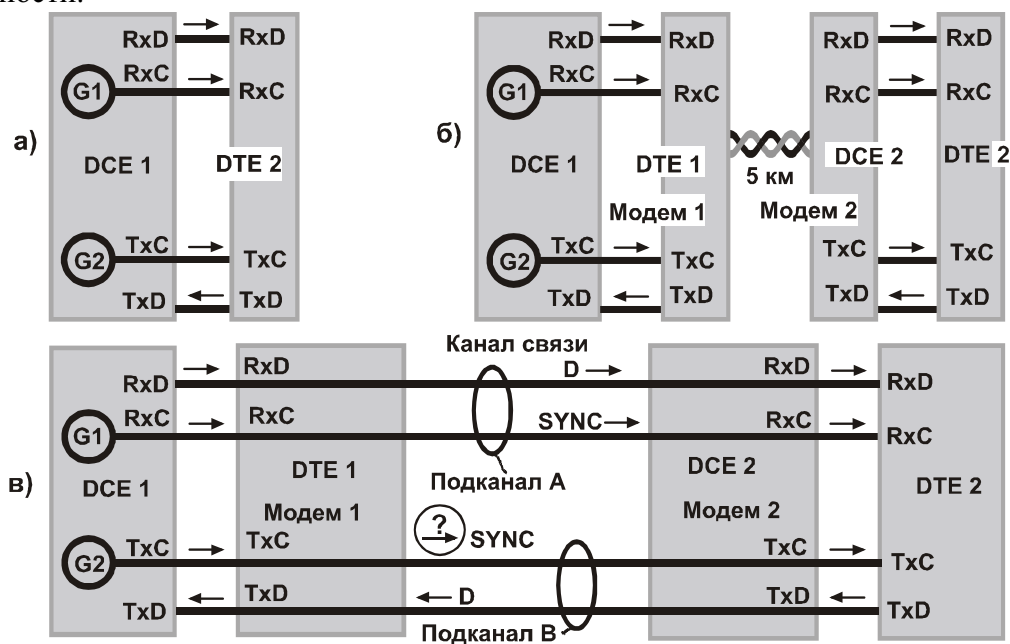


рис. 3.35. Схема взаимодействия устройств DCE 1 и DTE: *а* – при непосредственном соединении; *б, в* – при соединении через удлинитель, образованный цепью “модем 1 – модем 2”

Поясним сказанное. Абстрагируясь от деталей, можно перейти от схемы рис. 3.35, *б* к эквивалентной схеме рис. 3.35, *в*. На ней укрупненно показаны четыре трассы распространения данных и синхросигналов. Канал связи (физически это витая пара проводов) логически представлен двумя подканалами А и В. Подканал А передает данные D и синхросигналы SYNC слева направо, и это не вызывает вопросов, поскольку мы знаем, что линейный передатчик модема 1 (2), как обычно, формирует смесь данных и синхроимпульсов и транслирует ее в подканал.

Однако подканал В должен работать довольно своеобразно. Данные, как и положено, передаются справа налево, но синхросигналы, согласно нашему замыслу (построению удлинителя), должны передаваться встречно! Этот факт отмечен на рисунке вопросительным знаком. В общем, “этого не может быть никогда”.

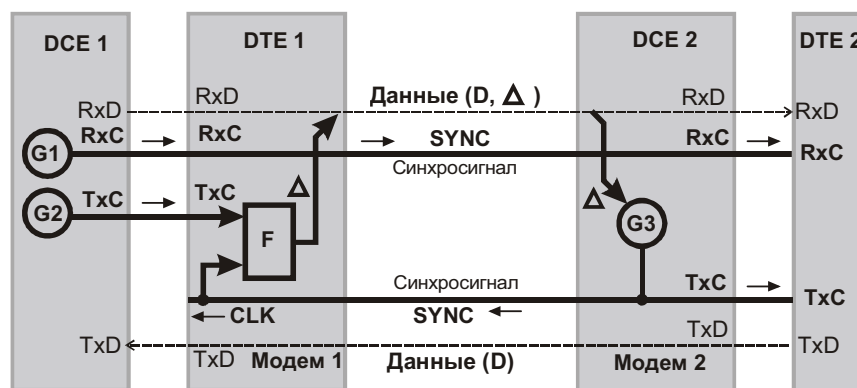


рис. 3.36. Детализация схемы, приведенной на рис. 3.35, *в*.

Решение задачи приведено на рис. 3.36. Как и в предыдущем примере (см. п . 3.5.1), применена канальная петля обратной связи, управляемая фазовым компаратором F; модемы 1 и 2 формируют кадры, содержащие как полезные, так и служебные данные. Код Δ разбаланса фаз передается по каналу связи и воздействует на фазу выходного сигнала высокостабильного генератора G3, номинальная частота которого равна частоте генератора G2 (реально эти частоты незначительно различаются).

В результате такого воздействия разбаланс фаз поддерживается на допустимо низком уровне, сигналы на входах компаратора совпадают с достаточной точностью, генератор G3 “привязан” по частоте к генератору G2, что, в сущности, и требовалось. Иными словами, можно полагать, что сигнал с генератора G2 неким образом проходит через удлинитель и с некоторой задержкой (что в любом случае неизбежно, но неважно) достигает входа TxС компьютера DTE 2. Так что вопросительный знак на рис. 3.35, *в* следовало бы заменить восклицательным!

Отметим, что схемы, приведенные на рис. 2.4 и рис. 3.34, также можно рассматривать как частные случаи удлинителей линий передачи сигналов TxС, TxD, RxC, RxD, в которых сигнал TxС распространяется “против течения” потока данных. Особенность этих схем в том, что сигналы TxС и RxC на стыках между размещенными рядом устройствами синхронны и синфазны.

3.6. Взаимодействие удаленных устройств с непосредственной односторонней передачей синхросигнала по каналу связи

Ранее (см. рис. 2.4 и рис. 3.34) были рассмотрены две схемы получения синхронных и синфазных потоков данных. Первая схема построена на основе буферной памяти, вторая использует фазовый компаратор и канальную петлю обратной связи. Здесь представлено еще одно, менее универсальное, но, возможно, более экономичное решение той же задачи (рис. 3.37) [59]. Его основное отличие от предыдущих состоит в применении трех витых пар проводов в канале связи. Одна пара используется для передачи синхросигналов, по двум другим транслируются данные.

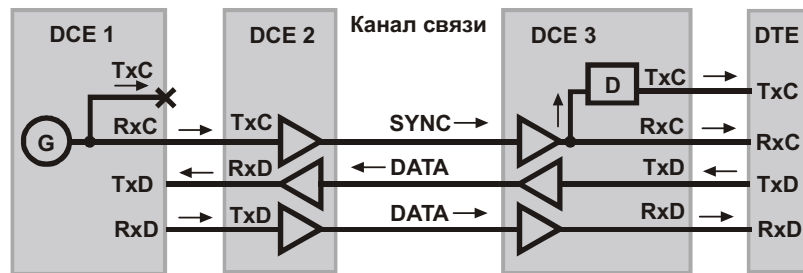


рис. 3.37. Система с непосредственной передачей синхросигнала по каналу связи: D – элемент задержки; выход TxC устройства DCE 1 не используется

Напомним, в чем заключается задача. На практике встречается ситуация, при которой устройство типа DCE (в нашем примере – устройство DCE 1) генерирует совпадающие синхросигналы RxC и TxC. Сигнал RxC сопровождает данные, выдаваемые из устройства DCE 1. Сигнал TxC задает временную сетку для принимаемых данных. Источником этих данных является устройство DTE, которое удалено от устройства DCE 1 на значительное расстояние. Поэтому нужно каким-то образом “заставить” устройство DTE подстроиться так, чтобы передаваемые им данные, подойдя к устройству DCE 1, попали в нужную временную сетку. В данном решении это достигается введением в устройство DCE 3 элемента D регулируемой фазовой задержки синхросигнала. Рассмотрим работу системы (рис. 3.37).

При передаче данных “слева – направо” они проходят из устройства DCE 1 в устройство DTE по нижней трассе. Синхросигнал от генератора G передается по верхней трассе с такими же физическими параметрами, поэтому он задерживается в той же мере, что и данные. Так что на входах RxD и RxC устройства DTE сигналы находятся в правильных фазовых соотношениях (см. рис. 1.8). (В действительности некоторый перекося сигналов, конечно, существует, но при достаточно низкой скорости передачи данных им можно пренебречь.)

Данные выдаются из устройства DTE под управлением сигнала на его входе TxC (см. рис. 1.6, б). При этом не имеет значения, в каком фазовом соотношении находятся сигналы на его входах TxC и RxC. Изменяя это соотношение, можно перемещать данные на входе устройства DCE 1 относительно синхросетки сигнала с генератора G. В частности, при соответствующей настройке элемента D можно достичь правильного фазового соотношения между сигналами на входе TxD и выходе TxC устройства DCE 1.

Элемент D регулируемой фазовой задержки синхросигнала должен обеспечивать плавный или ступенчатый сдвиг фазы выходного сигнала относительно входного в диапазоне от нуля до 360 градусов. Требуемая фазовая задержка зависит от длин и типов кабелей между устройствами и элементной базы устройств DCE 2 и DCE 3.

Предположим, что некая фирма-производитель в массовом порядке поставляет потребителям системы, показанные на рис. 3.37, причем они могут отличаться только длиной кабеля, образующего канал связи. Тогда требуемая фазовая задержка элемента D является функцией одной переменной – длины этого кабеля. Вид функции нетрудно получить расчетным путем или моделированием и подтвердить экспериментально. Если эти работы проведены, то фирма-производитель при изготовлении кабеля определенной длины может закодировать необходимую величину фазовой задержки, например в виде паяных перемычек на соединителе, как показано на рис. 3.38.

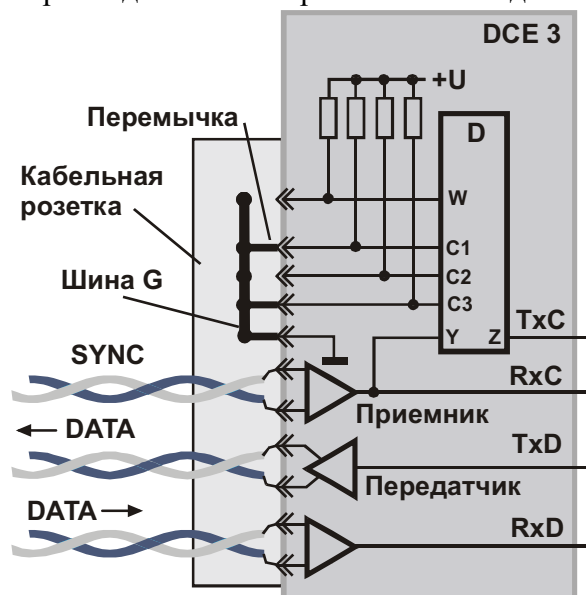


рис. 3.38. Схема сопряжения устройства DCE 3 с каналом связи

В данном примере соединитель содержит шесть пар контактов для подключения трех витых пар проводов кабеля, по которому передаются синхросигналы и данные. Кроме того, предусмотрены пять пар контактов для задания четырехразрядного кода, определяющего необходимую фазовую задержку синхросигнала. Один из контактов предназначен для подключения шины G кабельной розетки к шине нулевого потенциала устройства DCE 3. Остальные контакты розетки могут быть либо соединены, либо не соединены с шиной G с помощью индивидуальных паяных перемычек.

Если перемычка установлена, то на соответствующем управляющем входе элемента задержки присутствует сигнал лог. 0, в противном случае – сигнал лог. 1. Структура элемента задержки представлена на рис. 3.39, временные диаграммы преобразования сигналов – на рис. 3.40.

Если перемычка установлена, то на соответствующем управляющем входе элемента задержки присутствует сигнал лог. 0, в противном случае – сигнал лог. 1. Структура элемента задержки представлена на рис. 3.39, временные диаграммы преобразования сигналов – на рис. 3.40.

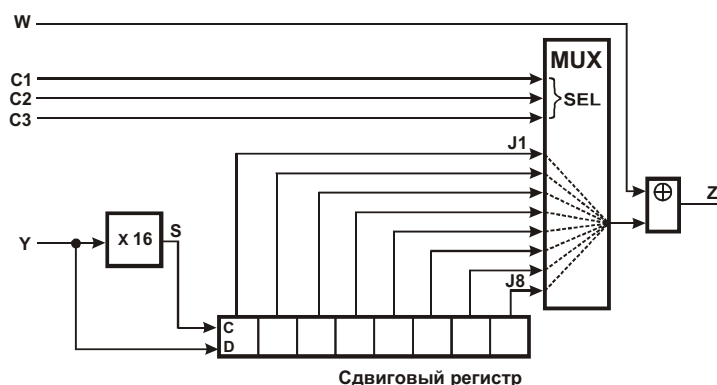


рис. 3.39. Структура элемента задержки D

Элемент задержки содержит умножитель частоты, сдвиговый регистр, мультиплексор и логический элемент Исключающее ИЛИ. Частота входного синхросигнала Y умножается на 16 и используется для продвижения информации в сдвиговом регистре.

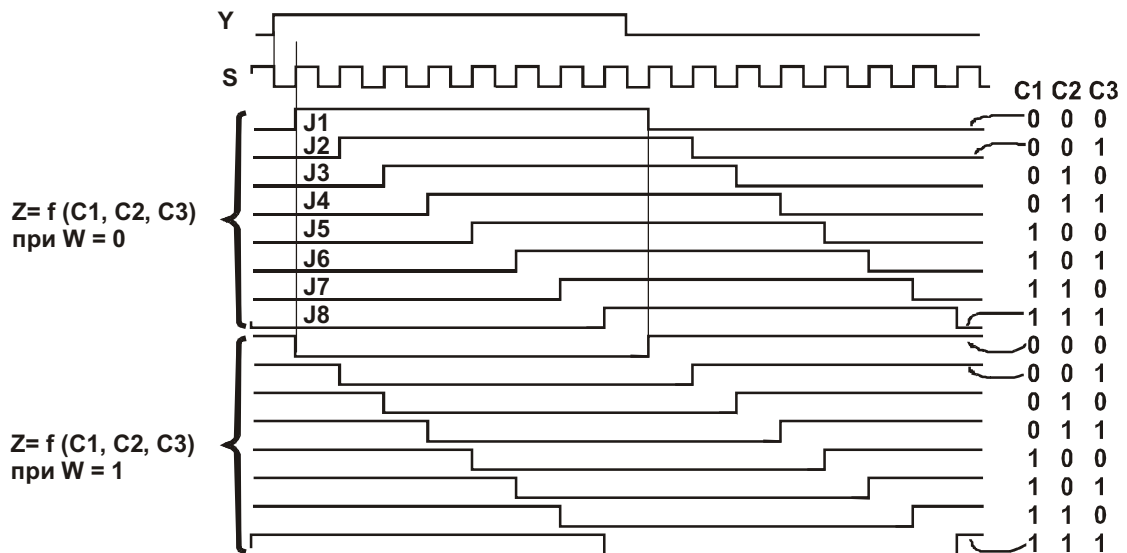


рис. 3.40. Временные диаграммы работы элемента задержки D

В зависимости от кода $C1 C2 C3$ один из выходных сигналов сдвигового регистра транслируется через мультиплексор и логический элемент Иключающее ИЛИ на выход Z . При этом сигнал инвертируется, если $W = 1$. Таким образом, шаг фазовой задержки составляет $1/16$ часть периода входного синхросигнала или $360/16 = 22,5$ градуса.

Умножитель частоты может быть выполнен по схеме, показанной на рис. 3.41.

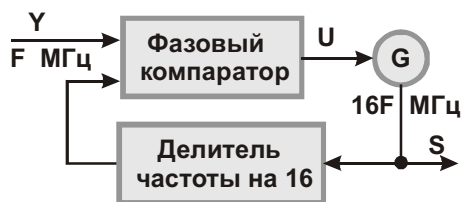


рис. 3.41. Схема умножителя частоты

Генератор G формирует сигнал S частотой $16F$, где F – частота входного сигнала Y . Фазовый компаратор следит за совпадением фронтов входных сигналов. При обнаружении более или менее устойчивого разбаланса фронтов он корректирует напряжение U управления частотой генератора в направлении улучшения совпадения. Поэтому сигнал S привязан по фазе к сигналу Y . В данном примере предполагается, что положительный фронт сигнала Y совпадает с отрицательным фронтом сигнала S (см. рис. 3.40).

Таким образом, применение отдельной витой пары проводов для передачи синхросигнала позволяет упростить схемные решения по сравнению с приведенными ранее (см. рис. 2.4 и рис. 3.34).

3.7. Поиск начала асинхронного сообщения

Ранее (см. рис. 1.5) мы рассмотрели асинхронный способ передачи данных между устройствами. При такой передаче данные представлены потоком символов, каждый из которых снабжен служебными битами Старт и Стоп.

Напомним, что фронт старт-бита представляет собой метку времени, от которой приемник отсчитывает длительности битовых интервалов данного символа. Стоп-бит разграничивает конец текущего и начало следующего символа, но между ними могут быть и достаточно длинные паузы. Число битов данных (5...8), наличие или отсутствие контрольного разряда (дополняющего число единичных битов символа до четного или нечетного) и скорость передачи задаются при начальной настройке взаимодействующих устройств.

Асинхронность состоит в том, что приемник не знает, в какой момент поступит очередной символ. Иными словами, отсутствует единая для всего потока данных синхронизация, т. е. она имеет локальный характер – устанавливается заново всякий раз при обнаружении приемником начала очередного старт-бита.

Здесь мы рассмотрим процесс установления синхронизации приемника с передатчиком на уровне передачи сообщений, т. е. групп символов [24]. Такие группы могут иметь переменную длину, например в диапазоне от 1 до 106 символов.

Как показано на рис. 3.42, сообщения разделены паузами (масштаб отображения нарушен для повышения наглядности рисунка; в действительности паузы могут быть не столь большими). В данном примере символы, составляющие сообщение, имеют следующую структуру: старт-бит, байт (D0 – D7) данных, стоп-бит и предшествующий ему признак первого (или не первого) символа в группе. Этот признак определен следующим образом: первому символу сообщения соответствует напряжение низкого уровня; последующим – напряжение высокого уровня.

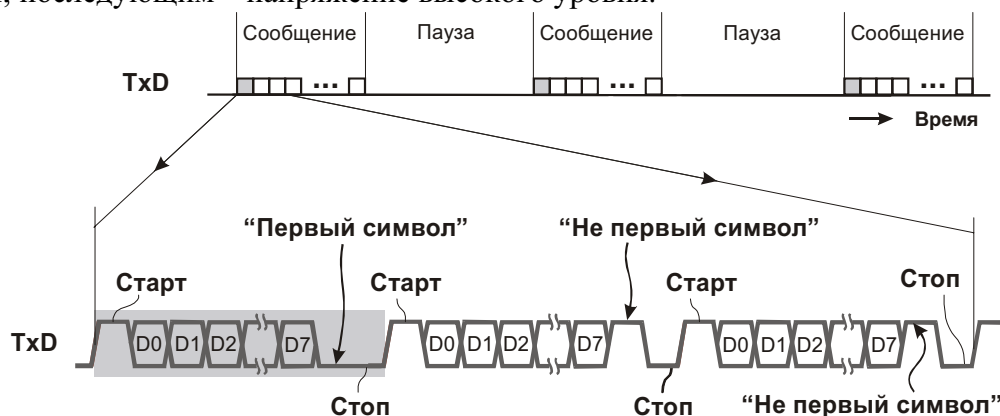


рис. 3.42. Передача асинхронных сообщений по последовательному каналу связи

Чтобы распознать начало сообщения, приемник анализирует состояние линии TxD. Пауза между сообщениями должна превышать некоторый заранее заданный интервал, например длительностью 10 мкс, поэтому на первом этапе поиска приемник дожидается ее появления (при этом паузы с меньшей длительностью не принимаются во внимание).

На втором этапе приемник анализирует полученный после паузы символ и убеждается в том, что он помечен признаком “Первый символ”. Если это не так, то констатируется ошибка. На последующих этапах проверяется длительность возможных пауз между символами (она не должна превышать, например, 8 мкс) и наличие признаков “Не первый символ”. Невыполнение этих условий рассматривается как ошибка передачи.

Такая многоэтапная проверка позволяет повысить надежность распознавания границ сообщений и обеспечить обнаружение ошибок.

3.8. Обнаружение и исправление ошибок синхронизации при передаче непрерывного асинхронного потока данных

При асинхронной передаче данных между устройствами используются биты Старт и Стоп, обозначающие начало и конец информационной посылки. Если такие посылки следуют вплотную друг к другу, то биты Старт и Стоп “прячутся” среди информационных битов, т. е. становятся внешне неотличимыми от них. Может случиться так, что приемник по каким-либо причинам потерял синхронизацию с передатчиком либо включился в работу в то время как плотный поток асинхронных посылок уже присутствует на входе устройства.

Чтобы отыскать положение стартовых битов в потоке данных, можно накопить и обработать статистику перепадов сигналов между битовыми интервалами. На границах между стоп- и старт-битами всегда должен наблюдаться отрицательный фронт сигнала. Но для статистического анализа нужен достаточный “интеллект” приемника и, возможно, большое время наблюдения, так как массив передаваемых данных может быть представлен повторяющимися посылками, например вида 1010101. В данном случае будут обнаружены регулярно повторяющиеся информационные перепады сигнала, неотличимые от стартовых. Нельзя ли упростить решение задачи вхождения приемника в синхронизм с передатчиком?

Сначала небольшое отступление. Как показано в примере на рис. 3.43, при неправильной синхронизации приемник расценивает бит 2 кода 1000110 как стартовый (ST). Бит 3 передаваемых данных воспринимается как бит 1, бит 4 – как бит 2 и т. д.



Рис. 3.43. Восстановление правильной синхронизации без обнаружения ошибки. ST и SP – старт- и стоп-биты информационных посылок

Однако благодаря передаче кода 1111111 правильная синхронизация восстанавливается. Действительно, с точки зрения приемника этот код неотличим от обычной паузы между посылками. Последующий отрицательный фронт сигнала расценивается (на этот раз справедливо) как начало стартового бита очередной информационной посылки. Можно убедиться в том, что передача кода 1111111 гарантирует восстановление синхронизации при любом возможном смещении посылки из-за несвоевременного старта.

Отметим, что приемник может “не знать”, что он работал неправильно, если не зафиксированы ошибочные (нулевые) значения битов Стоп (SP). В процессе перехода к правильной синхронизации один из информационных битов кода 1111111 рассматривается приемником как стоп-бит, имеющий правильное логическое состояние. Поэтому момент перехода к правильной синхронизации не регистрируется, что усложняет последующую локализацию ошибок протоколами более высоких уровней.

В рассмотренном далее решении [62] для восстановления правильной синхронизации и обнаружения этого события используется периодическая передача служебного кода 0000000 (рис. 3.44). Этот код передается, например, в каждой тысячной посылке. При этом соблюдаются определенные правила, позволяющие отличить служебный код от обычного нулевого кода данных.

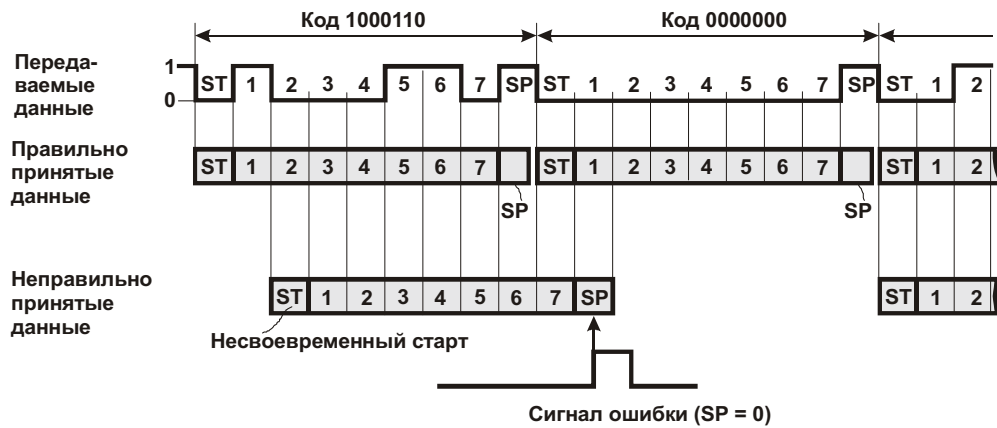


рис. 3.44. Восстановление правильной синхронизации с обнаружением ошибки

Из рисунка видно, что код 0000000, так же как и код 1111111, применим для восстановления правильной синхронизации. Однако концовка неправильно принятой посылки формируется при взаимодействии с кодом 0000000 и поэтому будет обязательно содержать ошибочный стоп-бит ($SP = 0$). Это позволяет сформировать сигнал ошибки или его программный аналог. Таким образом, передатчик может периодически вводить в передаваемый поток данных служебные нулевые коды, предназначенные для обнаружения и коррекции возможных ошибок синхронизации приемника.

Чтобы отличить служебный нулевой код от нулевых данных, можно воспользоваться одним из следующих правил.

1. Нулевые данные (в отличие от служебного кода) дублируются. Так, вместо кода данных 0000000 передается пара кодов 0000000, 0000000. Приемник выделяет парные нулевые коды, отбрасывает дубли и рассматривает оставшиеся коды как данные. Непарные нулевые коды отбрасываются как служебные.

2. В каждую посылку, как обычно, вводится разряд контроля по четности (нечетности). Нулевые данные не дублируются. Они сопровождаются правильными контрольными разрядами. В отличие от нулевых данных, служебные нулевые коды преднамеренно снабжаются неправильными контрольными разрядами. Приемник игнорирует эти коды как не содержащие полезной информации.

Можно предложить и иные способы разделения служебных и полезных данных для более надежной синхронизации в условиях повышенного уровня помех.

3.9. Распознавание межбайтовых границ в непрерывном синхронном потоке данных

В непрерывном синхронном потоке данных должны быть выделены какие-либо ориентиры для распознавания структурных единиц: кадров, ячеек, пакетов и т. п. Так, для распознавания межкадровых границ в синхронный поток данных обычно вводят группы служебных битов – флаговые коды (см гл. 7). Кадр может содержать сотни и тысячи битов, а его флаговый код может быть представлен двумя байтами. При таком соотношении числа служебных (флаговых) и полезных битов данных накладные расходы невелики, но с уменьшением длины кадра они растут.

В рассмотренных далее решениях каждый кадр переносит небольшое число битов полезных данных. Разграничение байтов с помощью многобитных флаговых кодов в данном случае нецелесообразно из-за плохого соотношения числа служебных и полезных битов кадра. Предлагаются два более эффективных способа разграничения байтов. Первый способ (п. 3.9.1) состоит в том, что в качестве флагового используется инвертированный бит данных. Второй способ (п. 3.9.2) основан на преобразовании байта в

20-разрядный кадр. Это позволяет ускорить процесс установления синхронизации приемника с передатчиком и обнаруживать ошибки при передаче данных.

3.9.1. Схема с дублированием и инвертированием одного бита данных

В системе передачи данных [27] передатчик (рис. 3.45) преобразует непрерывный поток байтов в непрерывный поток битов. Байт содержит биты DATA IN 1 – DATA IN 9, например, восемь информационных и один контрольный. Очередной байт параллельно загружается в сдвиговый регистр RG и затем последовательно выдвигается из него в линию связи через усилитель (усилитель на рисунке не показан). При этом для разграничения байтов выходного потока данных SDATA применен метод, во многом схожий с известным методом формирования асинхронных информационных посылок, когда они обрамляются служебными битами Старт и Стоп (см. п.1.2).

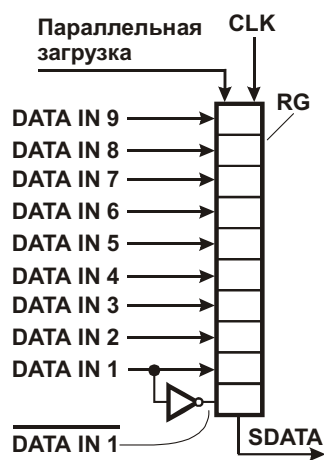


рис. 3.45. Схема формирования потока данных SDATA передатчиком

Данные из сдвигового регистра выдвигаются в такой последовательности. Сначала в линию поступает инвертированный бит $\text{NOT}(\text{DATA IN } 1)$, затем этот же бит DATA IN 1 без инверсии. Вслед за ним выдвигаются остальные биты DATA IN 2 – DATA IN 9. Таким образом, начало передачи байта сопровождается гарантированным изменением уровня сигнала. Подобно переднему фронту бита Старт в асинхронной посылке это изменение уровня сигнала используется для подстройки синхронизации между приемником и передатчиком.

Отметим, что здесь подстройка синхронизации ведется приемником по переднему фронту первого информационного бита, а не по переднему фронту бита Старт. Иными словами, момент подстройки вплотную приближен к первому полезному биту данных, а не удален от него на один бит, как в асинхронной посылке. Это повышает точность построения синхросетки приемником. Еще одно преимущество: отсутствует бит Стоп. (Этот бит был введен в асинхронную посылку в качестве антипода бита Старт для создания гарантированного флагового перепада уровней сигнала в линии.)

Схема преобразования последовательного потока данных в параллельный (рис. 3.46), размещенная в приемнике, содержит регистры RG 1 и RG 2, а также генератор и формирователь синхросигналов. Регистр RG 1 состоит из десяти D-триггеров с индивидуальными входами C синхронизации. Бит данных с входа D заносится в триггер по положительному фронту сигнала на входе C. Регистр RG 2 принимает девятиразрядный код по положительному фронту на общем входе C синхронизации.

В установленном режиме (когда достигнута устойчивая синхронизация приемника с передатчиком) данные SDATA последовательно записываются в триггеры регистра RG 1. После заполнения этого регистра данные переписываются в регистр RG 2, на выходах которого формируется поток байтов “DATA OUT”, эквивалентный исходному потоку “DATA IN”. Особая роль в поддержании синхронизации между передатчиком и приемником отведена выделенному на рисунке серым фоном балансному триггеру регистра RG 1. Он обеспечивает привязку фронтов синхросигнала от внутреннего генератора к центрам битовых интервалов входного сигнала SDATA. Процесс приема и преобразования данных поясняется временными диаграммами, приведенными на рис. 3.47.

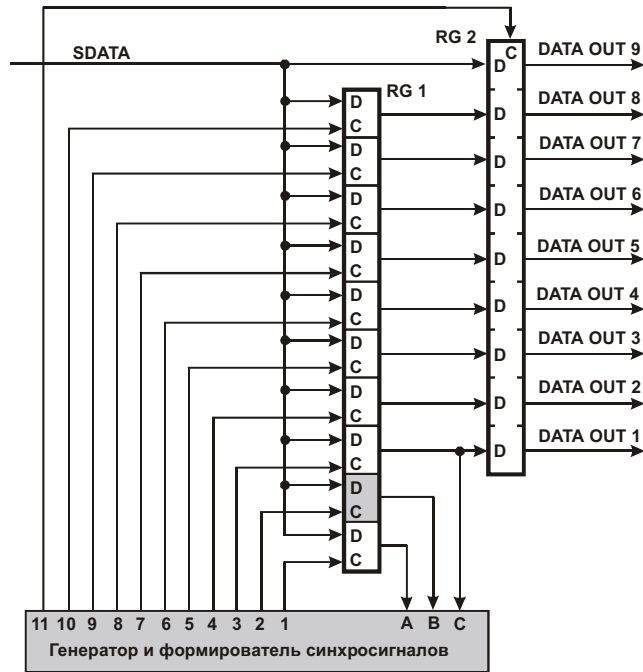


рис. 3.46. Схема преобразования приемником последовательного потока данных в параллельный

Внутренний генератор приемника формирует непрерывный периодический сигнал G . В данном примере его частота в два раза превышает частоту сигнала CLK и может регулироваться в небольших пределах. Сигналы 1 – 11 формируются из сигнала G и поэтому привязаны к нему. Положительные фронты сигналов 1, 3 – 11 соответствуют центрам битовых интервалов принимаемых данных. Положительный фронт сигнала 2 в идеальном случае совпадает с положительным или отрицательным фронтом сигнала переключения между инвертированным и прямым битами $SDATA$ 1. При этом в балансном триггере с равной вероятностью может быть зафиксирован лог. 0 или лог. 1.

Если синхронизация уже достигнута, то она поддерживается следующим образом. В каждом цикле приема байта сразу после обновления сигналов A , B и C они анализируются. Основное условие правильной синхронизации – противоположность логических значений сигналов A и C . При его выполнении проверяется своевременность формирования сигнала записи бита в балансный триггер. Если $B = A$, т. е. если запись бита данных в балансный триггер произошла слишком рано, то частота сигнала G чуть уменьшается. При $B = C$, т. е. в ситуации, когда запись бита данных в балансный триггер произошла слишком поздно, частота этого сигнала в той же мере очень незначительно увеличивается.

В результате такой регулировки положительный фронт сигнала 2 на входе синхронизации балансного триггера постоянно и в небольших пределах колеблется около теоретического положения, соответствующего началу битового интервала $SDATA$ 1. При этом положительные фронты сигналов 1, 3 – 11 в среднем во времени точно совпадают с центрами битовых интервалов.

Если в исходном состоянии синхронизация не достигнута, то частота сигнала G не корректируется. В отсутствие коррекции эта частота не может точно соответствовать удвоенной скорости передачи данных по линии и отличается от нее, например, на несколько герц. Разность частот приводит к медленному дрейфу группы сигналов G , 1 – 11 относительно “неподвижного” сигнала $SDATA$.

По мере дрейфа синхросетки положительный фронт сигнала 2 последовательно и достаточно медленно проходит через границы между битами. Одна из этих границ –

искомая. Она соответствует началу байта. Остальные границы разделяют информационные биты внутри байта.

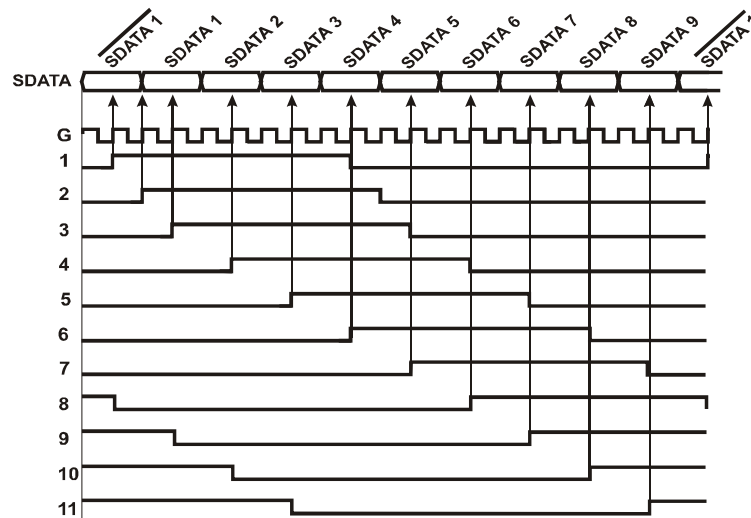


Рис. 3.47. Временные диаграммы преобразования последовательного потока данных в параллельный

Чтобы отличить искомую границу от остальных, необходим некоторый “интеллект” приемника. Прохождение искомой границы характеризуется следующими признаками.

1. При подходе к границе сигналы А и С устойчиво (статистически достоверно) противофазны. Сигнал В устойчиво совпадает с сигналом А (или С в зависимости от направления дрейфа синхросетки).

2. После прохождения границы сигналы А и С остаются устойчиво противофазными, но сигнал В теперь устойчиво совпадает с сигналом С (или А в зависимости от направления дрейфа синхросетки).

После обнаружения первого, а затем и второго условия приемник приостанавливает дрейф синхросетки путем коррекции частоты сигнала G в нужном направлении, как было описано ранее. Временная диаграмма приходит в соответствие показанной на рис. 3.47.

Можно несколько усложнить схему и применить более интеллектуальные способы установления и поддержания синхронизации. Например, вместо одного балансного триггера можно ввести группу триггеров для более точной “ориентации во времени” и использовать более мелкую временную сетку. При этом частоту (фазу) сигнала G можно не корректировать, если она находится в допустимых пределах. Возможны также разные виды реакции на обнаружение ошибок.

Конечно, не исключено ложное опознание границ байтов, если передаваемая информация регулярна и содержит постоянный перепад сигналов по крайней мере в одной паре соседних информационных битов. Но, вероятнее всего, передача такого специфического информационного блока не может быть “бесконечной”. Рано или поздно ошибка определения межбайтовых границ обнаружится с последующим установлением правильной синхронизации. (Подобное справедливо и по отношению к непрерывному потоку асинхронных посылок с битами Старт и Стоп.) Напомним, что для разрушения закономерностей в передаваемой информации применяют скремблирование (см. п. 8.4).

3.9.2. Схема с дублированием и инвертированием байта данных

Для распознавания межбайтовых границ в непрерывном синхронном потоке данных можно применить схему кодирования, приведенную на рис. 3.48 [68].

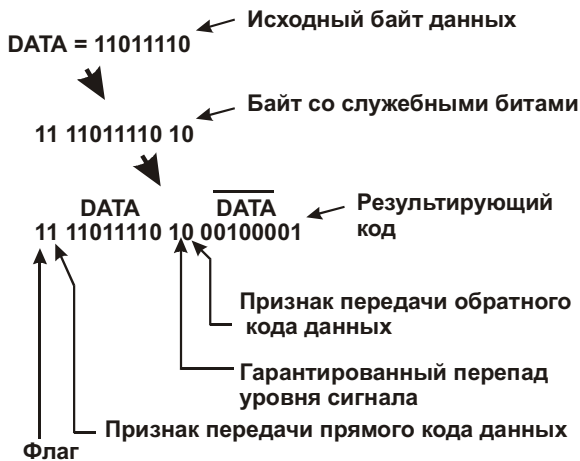


Рис. 3.48. Процесс преобразования байта данных для его передачи по линии

К исходному байту данных, в данном примере равному 11011110, добавляются две пары служебных битов: 11 и 10. Затем к полученному 12-разрядному коду добавляется инвертированный байт данных. Полученный 18-разрядный код передается в линию.

Приемник непрерывно просматривает поступающий поток данных. Схема или программа анализа данных настроена на

выявление закономерности, принятой при кодировании. Рано или поздно приемник обнаруживает, что биты, разделенные девятью тактами, совпадают с точностью до “наоборот”, но один из них (флаговый) остается неизменным. С учетом того, что вслед за флаговым битом должна следовать лог. 1 (признак передачи прямого кода данных), приемник окончательно распознаёт положение межбайтовых границ.

Комбинация служебных битов “10₂” обеспечивает гарантированный перепад уровней сигнала, который необходим для синхронизации генератора приемника. Дублирование данных позволяет обнаруживать ошибки передачи при сравнении прямого и обратного кодов. Если ввести дополнительный бит контроля по четности (нечетности), то можно корректировать одиночные ошибки передачи.