# Вероятностная синхронизация в телекоммуникационных системах: мониторинг уровня ошибок в линии связи

*Одна из основных характеристик линии связи (проводной, радиорелейной, оптоволоконной и т. п.) – уровень ошибок в потоке передаваемых по ней данных. Ошибки в принимаемых данных могут быть вызваны, в основном, факторами логического или физического уровня. К первым относятся потери пакетов данных из-за информационной перегрузки приёмника, ко вторым – перекрёстные помехи, отражения сигнала от неоднородностей линии связи и т. п. В статье рассмотрены встроенные в систему передачи данных средства обнаружения ошибок физического уровня. Предложен способ их мониторинга и схемные решения, уменьшающие вводимую в поток полезных данных избыточность благодаря применению вероятностной синхронизации [1].*

Обычно для обнаружения ошибок физического уровня применяют BER-тестеры [2], которые генерируют потоки псевдослучайных битов, собранных в пакеты или выдаваемых в линию без какого-либо разделения на структурные единицы (ячейки, кадры и т. п.). В любом случае тестовые данные распространяются по линии связи в одну или обе стороны (по петле или в виде независимых встречных потоков). Ошибки контролируются на дальней или (и) ближней сторонах линии. Такой способ контроля частично или полностью прерывает передачу полезной информации по линии связи и поэтому малоэффективен при мониторинге ошибок на фоне штатного режима работы системы.

Можно было бы оценивать уровень ошибок по интенсивности отбраковки пакетов полезных данных приёмником, однако, как уже упоминалось, причиной потерь пакетов может быть перегрузка приёмника, а не низкое качество передачи сигналов по линии связи. Кроме того, не все технологии передачи данных предусматривают генерацию и проверку контрольных сумм (кодов CRC) при компоновке передаваемых и анализе принимаемых пакетов. Например, часто не контролируются пакеты, переносящие звуковую или видеоинформацию в реальном времени.

Таким образом, желательно построить систему передачи произвольных пакетов данных, даже не содержащих контрольных сумм, со встроенным монитором ошибок и практически нулевой избыточностью, вносимой в полезный информационный поток.

## 1. Функциональная схема предлагаемой системы передачи данных

Общая идея построения предлагаемой системы передачи данных (рис. 1) состоит в следующем. Непрерывный поток полезных (а не тестовых) скремблированных данных в линии связи “нарезается” на кадры Qi (рис. 2) импульсами J (J*) вероятностной синхронизации. Эта нарезка никак не связана с границами передаваемых по линии пакетов полезных данных.

Передающее устройство (A) дополняет каждый кадр Qi контрольной суммой (скремблированным кодом CRC), которая размещается в начале следующего кадра Q(i + 1). Приёмное устройство (B) проверяет контрольные суммы и накапливает статистические данные о соотношении времени работы с ошибками и времени безошибочной работы. В дальнейшем накопленные статистические данные могут быть считаны из устройства B, например с использованием протокола сетевого администрирования SNMP (процедура считывания этих данных здесь не рассматривается).

Прежде чем перейти к более подробному описанию работы системы, рассмотрим её составные части. В данном примере построения системы показан только один канал передачи данных (слева направо). Канал передачи данных в обратном направлении строится аналогично.

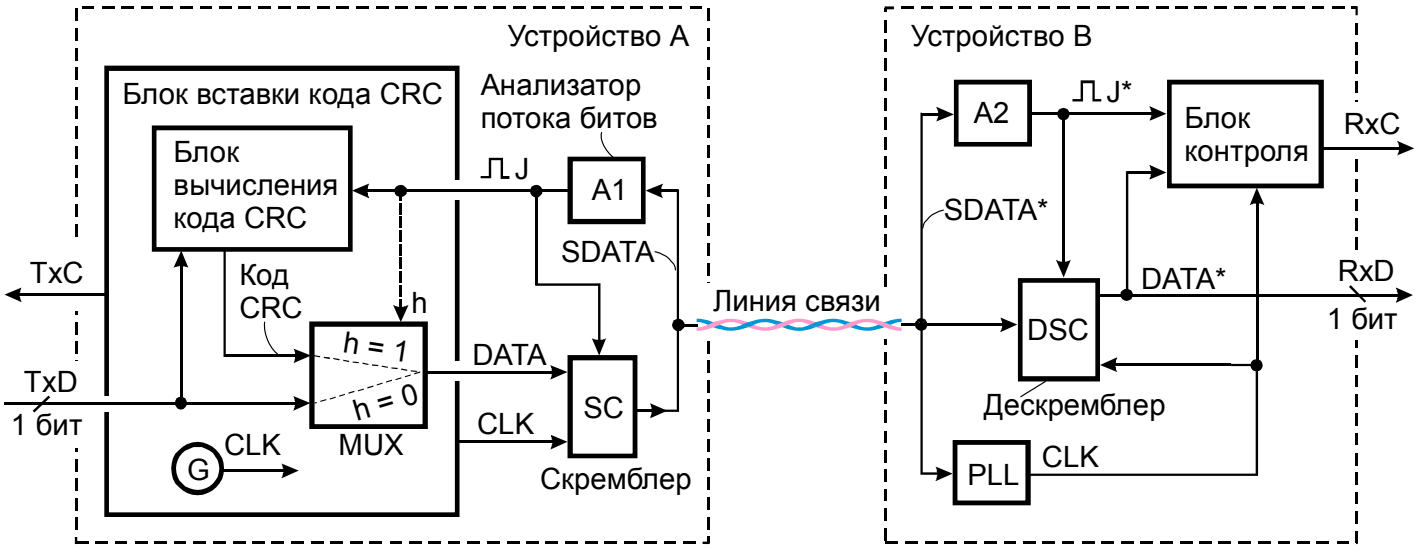


Рис. 1.↑ Предлагаемая система передачи данных со встроенным монитором ошибок в линии связи (первый вариант)

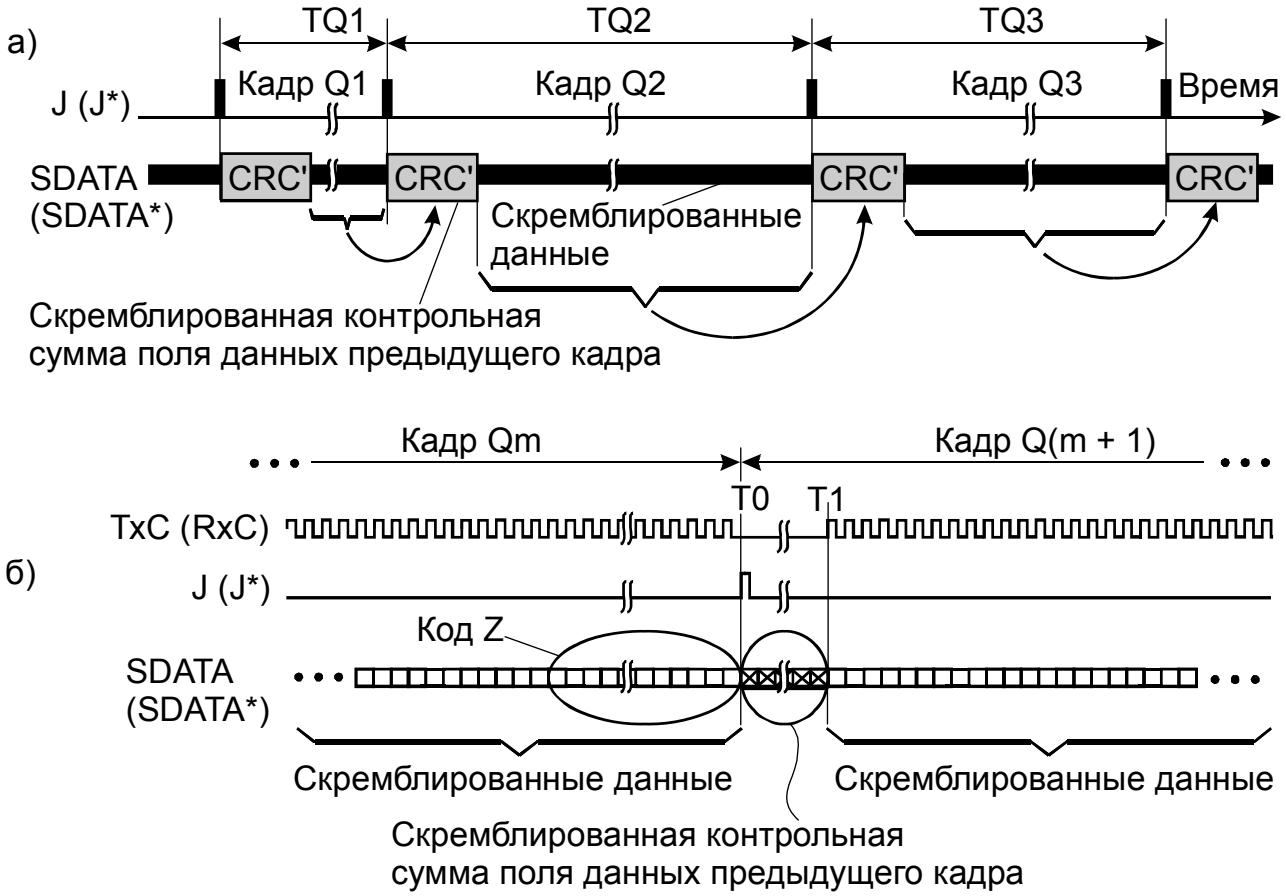


Рис. 2. Укрупнённая (*а*) и более детальная (*б*) диаграммы работы системы передачи данных (рис. 1)

Устройство A предназначено для скремблирования и передачи полезных данных TxD с входа системы в линию связи (данные SDATA). Всякий раз при получении импульса J вероятностной синхронизации передача данных TxD прерывается, например на 16 тактов, в течение которых в линию передаётся только что вычисленная и прошедшая через скремблер SC контрольная сумма поля данных предыдущего кадра (16-разрядный код CRC').

Устройство B выполняет обратные операции. Принятый из линии поток битов SDATA* дескремблируется и передаётся на выход системы в виде потока полезных данных RxD. Истинность каждого бита подтверждается синхроимпульсом RxC. Всякий раз при получении импульса J* вероятностной синхронизации передача потока синхроимпульсов RxC прерывается на 16 тактов, так что внешнее устройство (на рисунке не показано) не воспринимает контрольную сумму, которая в этот период переносится сигналом RxD.

Устройство A содержит генератор G синхросигнала, мультиплексор MUX, скремблер SC, анализатор A1 потока битов и блок вычисления кода CRC. Устройство B содержит генератор PLL (Phase Lock Loop) с фазовой автоподстройкой частоты, дескремблер DSC, анализатор A2 потока битов и блок контроля.

Генератор G формирует синхросигнал – непрерывную последовательность тактовых импульсов CLK, определяющую скорость передачи битов по линии связи. Каждый период синхросигнала соответствует передаче одного бита. Передача битов по линии связи никогда не приостанавливается даже в отсутствие полезных данных, когда паузы между информационными пакетами, поступающими на вход TxD, заполняются флаговыми последовательностями битов [2].

Мультиплексор MUX в зависимости от значения управляющего сигнала h длительностью 16 тактов передаёт на выход сигнал с верхнего или нижнего входа. При h = 0 DATA = TxD, на выход мультиплексора передаётся основной поток битов данных; при h = 1 на выход мультиплексора пересылается последовательный 16-разрядный код CRC – контрольная сумма ранее переданного поля данных кадра Qi.

Скремблер SC предназначен для преобразования данных DATA к виду, когда их можно рассматривать как случайные. Это означает, что вероятности появления в каждом такте скремблированных битов SDATA = 0 и SDATA = 1 одинаковы и не зависят от предыстории. Скремблирование достигается сложением по модулю два сигнала DATA с сигналом от изолированного генератора псевдослучайной последовательности битов на основе кольцевого сдвигового регистра (генератор на рисунке не показан). Этот генератор устанавливается сигналом J в некоторое заранее известное начальное состояние.

В такое же состояние по сигналу J* устанавливается аналогичный генератор дескремблера DSC в устройстве B. Таким образом, первая же пара "самопроизвольно" формируемых импульсов J и J* устанавливает правильную кодовую синхронизацию между устройствами A и B, если синхронизация была по каким-либо причинам потеряна.

Битовая синхронизация генераторов псевдослучайной последовательности скремблера и дескремблера осуществляется сигналом CLK. В каждом такте работы системы на выходах этих генераторов формируются одинаковые псевдослучайные биты. Одноимённые такты работы устройств A и B разнесены во времени, в основном, из-за задержки сигнала в линии, но последовательности происходящих кодовых событий одинаковы. Поэтому можно условно считать, что задержки вообще нет, и импульсы J и J* формируются одновременно.

Сигнал скремблированных данных SDATA передаётся в линию связи. Детальное описание применённых в системе скремблера SC и дескремблера DSC, выполняющего операцию восстановления исходных данных, приведено в [3].

Линия связи может быть выполнена на основе витой пары проводов, оптического волокна либо реализована с использованием иных сред передачи сигналов. Линия может со-

держать ряд ретрансляторов, поэтому задержка распространения сигнала между устройствами A и B может быть значительной и составлять десятки миллисекунд или более.

Скремблированный поток данных SDATA (SDATA*) просматривается анализатором A1 (A2) последовательности битов для отыскания в нём некоторого заранее заданного кода Z. Если разрядность этого кода равна M, то он обнаруживается в случайном потоке битов в среднем один раз в каждой цепи длиной $2^M$ бит. Например, при M = 30 длина цепи примерно равна $10^9$ бит; при скорости передачи данных по линии, равной 1 Гбит/с (когда применён оптоволоконный кабель), средняя частота сигнала J (J*) обнаружения кода Z на выходе анализатора A1 составляет примерно 1 Гц. Структура анализатора описана в [2, 3]. После обнаружения кода Z анализатор блокируется по входу, например, на 32 такта во избежание возможного наложения конкурирующих процессов подачи на скремблер данных с разных направлений.

Генератор PLL с фазовой автоподстройкой частоты предназначен для формирования высокостабильного синхросигнала CLK на основе непрерывного слежения за входным сигналом скремблированных данных SDATA*, поступающим из линии. Этот сигнал в данном примере представлен кодом NRZ [2]. Благодаря достаточной инерционности генератора сигнал на его выходе практически нечувствителен к “дрожанию фазы” сигнала SDATA* и иным его кратковременным искажениям, вызванным помехами в линии связи. Некоторые схемные решения таких генераторов приведены в [2].

К моменту поступления очередного импульса J на вход блока вычисления кода CRC в нём накоплена контрольная сумма битов поля данных предыдущего кадра. В момент поступления импульса J эта контрольная сумма (код CRC) запоминается и затем в течение 16 тактов пересылается через скремблер SC в линию связи. К моменту окончания 16-го такта блок вычисления кода CRC устанавливается в нуль и начинает накопление новой контрольной суммы, соответствующей битам поля данных текущего кадра.

К моменту поступления импульса J* в отсутствие ошибок в поле данных предыдущего кадра блок контроля устройства B накопил точно такой же код CRC. В момент поступления импульса J* этот код запоминается и затем в течение 16 тактов побитно сравнивается с кодом, пересылаемым из устройства A по линии связи. К моменту окончания 16-го такта блок контроля устанавливается в нуль и также начинает накопление новой контрольной суммы, соответствующей битам поля данных текущего кадра. При этом положительный (а в других циклах работы, возможно, отрицательный) результат сопоставления вычисленного и полученного из линии связи кодов CRC учитывается при накоплении статистических данных об ошибках.

Статистические данные могут отражать уровень ошибок за последнюю минуту, 10, 30 минут, час и т. д. в терминах отношения времени работы с ошибками ко времени безошибочной работы. При этом в малых интервалах, порядка минуты – десятка минут, можно учитывать индивидуальные длительности использованных кадров, воспользовавшись таймером, а в больших интервалах (порядка часов и более) можно пользоваться средней длительностью кадра. В приведенном ранее примере она составляет около 1 с.

Возвращаясь к этому примеру (скорость передачи данных 1 Гбит/с, разрядность кода Z – 30 бит, частота следования импульсов J (J*) вероятностной синхронизации – около 1 Гц), можно заключить, что вносимая в поток полезных данных избыточность, приведенная к интервалу времени, равному 1 с, составляет 16 бит / $10^9$ бит = $1{,}6 \times 10^{-8}$.

## 2. Сопряжение с другими системами

Для сопряжения предлагаемой системы передачи данных с другими системами, имеющими близкие, но не равные скорости передачи битов, необходимы блоки буферной памяти (для краткости – буферы) и средства регулирования уровня их заполнения (рис. 3).

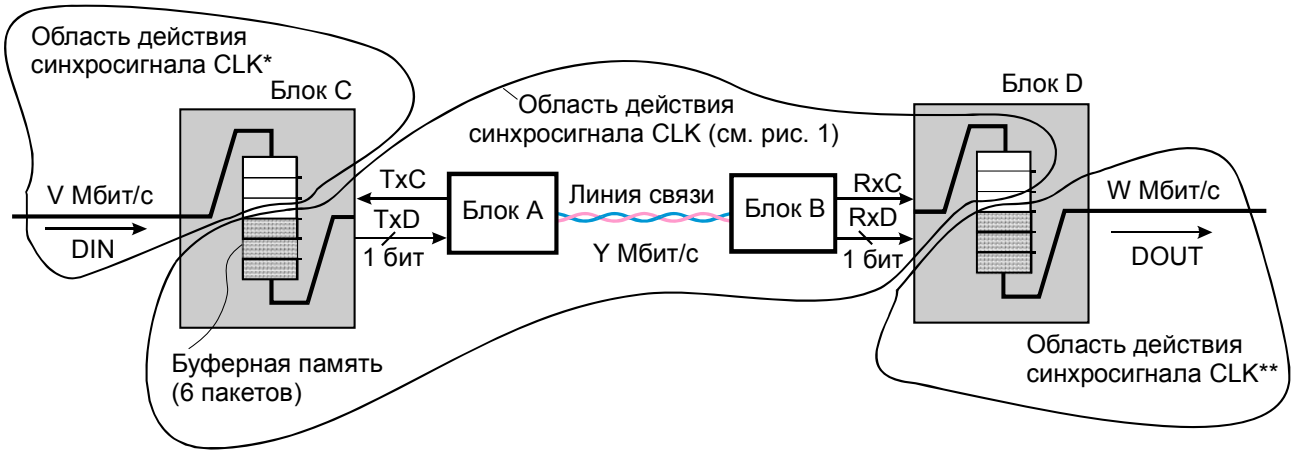


Рис. 3. Пример сопряжения предлагаемой системы с другими системами

Передаваемые данные сгруппированы в пакеты. Поддержание номинального уровня заполнения буферов (вблизи отметки “50%”) осуществляется управлением скоростями передачи пакетов (а не битов).

В примере, представленном на рис. 3, потоки полезных данных DIN и DOUT, собранных в пакеты, должны иметь одинаковые средние скорости, несмотря на то, что скорости передачи битов на разных участках транспортной системы немного, но устойчиво различаются. Так, битовый поток данных втекает из внешней линии связи в блок C со скоростью V Мбит/с под управлением синхросигнала CLK* от удалённого генератора (на рисунке не показан). Рассмотренная ранее (рис. 1) система синхронизируется сигналом CLK, который задаёт скорость Y Мбит/с передачи битов по линии связи между блоками A и B. Наконец, битовый поток данных вытекает из блока D во внешнюю линию связи со скоростью W Мбит/с под управлением некоторого синхросигнала CLK**. Как же осуществляется бесконфликтное сопряжение разноскоростных устройств?

Для упрощения описания предположим, что все передаваемые пакеты данных имеют одинаковую длину. Максимальная скорость передачи пакетов по линии связи достигается в том случае, когда они следуют друг за другом вплотную, без пауз. Скорость передачи пакетов уменьшается с введением пауз и увеличением их длительности (паузы заполнены флаговыми комбинациями битов [2]). В предельном случае скорость передачи пакетов может быть нулевой, в то время как скорость передачи битов постоянна и определяется частотой синхросигнала.

Предположим, что из-за различий частот синхросигналов максимальные скорости передачи пакетов (без пауз) по трём последовательно включенным линиям связи составляют (в разных сочетаниях) 99, 100 и 101 пакетов/мс. Максимальная пропускная способность такой последовательной цепи определяется самым медленным её звеном и в данном случае равна 99 пакетов/мс.

В процессе автоматического выбора максимально допустимой скорости передачи пакетов удалённое устройство, расположенное слева от блока C, постепенно наращивает скорость их выдачи, начиная с заведомо приемлемой величины, например, равной 90 пакетов/мс. Блоки C и D следят за тем, чтобы заполнение их буферов примерно соответствовало уровню 50 %. При повышении этого уровня блок C (D) увеличивает интенсивность считывания паке-

тов из буфера, (пока это возможно) и наоборот, при снижении уровня уменьшает темп выдачи пакетов увеличением пауз между ними.

Если устройство, расположенное слева от блока C, самое медленное в цепи (т. е. если его скорость V передачи битов соответствует одному из условий V < Y < W или V < W < Y), то оно способно генерировать не более 99 пакетов/мс, что укладывается в полосы пропускания других устройств (101 и 100 пакетов/мс). В этом случае устройство, расположенное слева от блока C, передаёт пакеты на максимально доступной для него скорости (без пауз между пакетами), в то время как блоки C и D для поддержания номинальных (50%) уровней заполнения буферов вынуждены вводить паузы между пакетами. В результате вся последовательная цепь из разноскоростных устройств передаёт данные с одинаковой скоростью, равной 99 пакетов/мс.

Если устройство, расположенное слева от блока C, не самое медленное в цепи, то оно способно генерировать более 99 пакетов/мс, что неприемлемо, по крайней мере, для одного из партнёров. Повышение скорости свыше 99 пакетов/мс приводит к потере пакетов из-за переполнения буфера самого медленного устройства. Информация о потере пакетов распространяется по обратному каналу (задача обнаружения потерь пакетов из-за перегрузки приёмника рассмотрена в [4]). В этом случае устройство, расположенное слева от блока C, снижает скорость выдачи пакетов и после экспериментального (методом проб и ошибок) поиска допустимой границы фиксирует её на уровне 99 пакетов/мс.

Попытки минимального повышения ранее достигнутой максимальной скорости передачи пакетов продолжаются и в установившемся режиме “в надежде” на то, что медленное устройство заменено более быстрым либо изменилась топология сети и т. п. Эти попытки могут следовать, например, с интервалом 100 с и в случае неудачи сопровождаются быстрым возвратом к достигнутому ранее значению скорости.

## 3. Использование встроенных служебных каналов связи

В системе передачи данных, показанной на рис. 1, применена встречная синхронизация (TxC – TxD [2]), причём поток полезных данных содержит разрывы, в течение которых по линии связи передаются группы служебных битов (контрольные суммы). Это создаёт некоторые неудобства при сопряжении предлагаемой системы с теми внешними устройствами, которые способны работать только с неразрывными потоками данных.

Для сопряжения с такими устройствами можно использовать встроенные служебные каналы связи, широко применяемые системах передачи данных [2]. Служебный канал организуется следующим образом. Данные передаются в виде фреймов (этот термин эквивалентен ранее использованным терминам “кадр”, “пакет” и используется здесь во избежание путаницы). Фрейм содержит флаговый код, служебные биты и поле данных.

В отличие от первого варианта системы (рис. 1) второй (рис. 4) предусматривает вставку кода CRC или его фрагмента в поле служебных данных ближайшего фрейма, который следует за импульсом J (J*) вероятностной синхронизации.

В данном случае внешний синхросигнал не прерывается, его направление совпадает с направлением сигнала TxD (применена “попутная синхронизация” [2]). Фреймы компонуются в устройстве A и расформировываются в устройстве B соответственно фреймером и дефреймером. Блок вычисления кода CRC и блок контроля, как и в первом варианте системы, непрерывно подсчитывают контрольные суммы, которые сверяются по сигналу J*. Средняя частота следования этого сигнала, как и ранее, может составлять 1 Гц. Непрерывный синхросигнал CLK проходит через устройство A, линию связи, устройство B и в виде сигнала RxC поступает на следующее устройство.

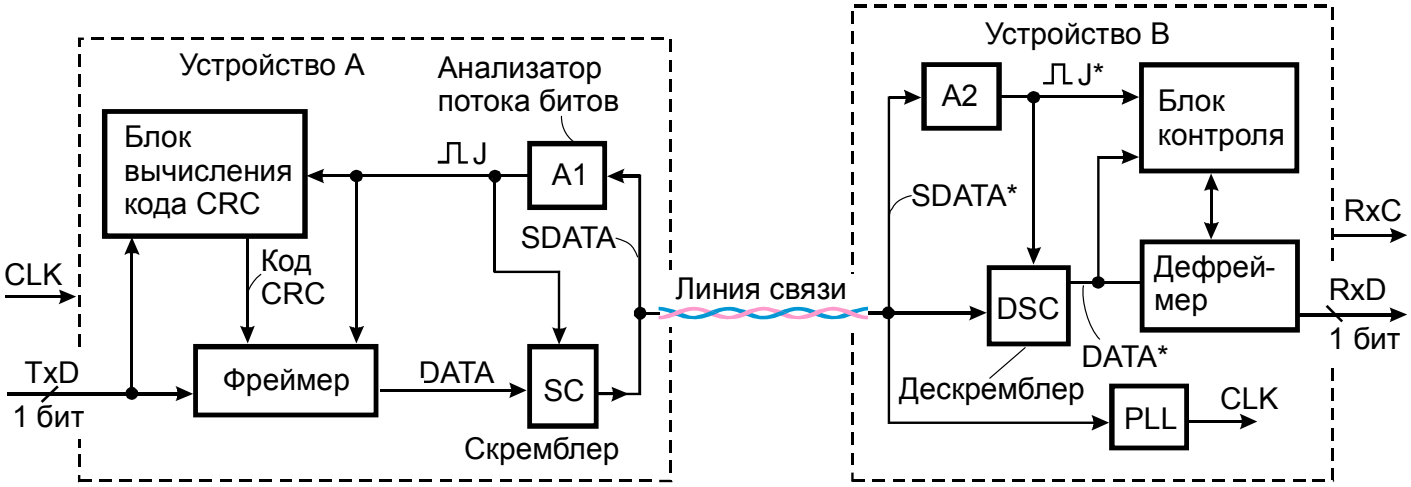


Рис. 4. Предлагаемая система передачи данных со встроенным монитором ошибок в линии связи (второй вариант)

## 4. Заключение

В статье предложены два варианта систем с мониторингом ошибок в потоке полезных (а не тестовых) данных. Полезные данные могут передаваться в составе пакетов, не контролируемых контрольными суммами. Тем не менее, ошибки контролируются на уровне “стохастических” кадров с заранее не предсказуемыми положениями границ – импульсов J (J*). Благодаря использованию вероятностной синхронизации [1], удалённые друг от друга устройства координируют действия по проверке этих кадров без введения в поток полезных данных каких-либо дополнительных средств их разграничения. Предлагаемые решения позволяют уменьшить объём служебной информации, передаваемой по каналам связи, до уровня порядка $10^{-8}$ и ниже, и вести непрерывный мониторинг ошибок физического уровня на фоне штатной передачи данных.

ЛИТЕРАТУРА:

1. *Б.В.Шевкопляс. Вероятностная синхронизация в телекоммуникационных системах: вставка команд в поток данных без использования избыточных битов — Схемотехника, 2005, № 5, с. 23 — 25, № 6, с. 23 — 26. Электронная версия статьи: http://lit.lib.ru/s/shewkopljas_b_w/*
2. *С.М.Сухман, А.В.Бернов, Б.В.Шевкопляс. Синхронизация в телекоммуникационных системах. Анализ инженерных решений. — М.: Эко-Трендз, 2003. Электронная версия книги: http://lit.lib.ru/s/shewkopljas_b_w/*
3. *Б.В.Шевкопляс. Скремблирование передаваемых данных — Схемотехника, 2004, №12, с. 24 — 27, 2005, №1, с. 29 — 32, 2005, №2, с. 32 — 35, 2005, №3, с. 30 — 33. Электронная версия статьи: http://lit.lib.ru/s/shewkopljas_b_w/*
4. *Б.В.Шевкопляс. Вероятностная синхронизация в телекоммуникационных системах: измерение параметров канала связи — Схемотехника, 2006, №9, с. 20 — 23 (часть 1), №10, с. 10 — 12 (часть 2). Электронная версия статьи: http://lit.lib.ru/s/shewkopljas_b_w/*

Б. Шевкопляс