

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Политехнический институт
Факультет приборостроения,
информационных технологий и
электроники

Специальность 10.05.02 -

Специализация -

Кафедра
«Информационная безопасность
систем и технологий»

Информационная безопасность
телекоммуникационных систем
Разработка защищенных
телекоммуникационных систем

ДИПЛОМНЫЙ ПРОЕКТ

на тему

УЗЛЫ ОБЕСПЕЧЕНИЯ ЦЕЛОСТНОСТИ ДАННЫХ В
АППАРАТУРЕ ПЕРЕДАЧИ ДАННЫХ СПЕЦИАЛЬНОГО НАЗНАЧЕНИЯ

Студент _____ Подгорецкая О.М.

Руководитель _____ Бессонов А.В.

Консультант _____ Кузнецова Г.А.

Нормоконтролер _____ Иванов А.П.

Рецензент ведущий инженер АО

«НПП «Рубин» _____ Будникова М.В.

Работа допущена к защите (протокол заседания кафедры от _____ № _____)

Заведующий кафедрой _____ С.Л.Зефилов

Работа защищена с отметкой _____ (протокол заседания ГЭК
от _____ № _____)

Секретарь ГЭК _____ А.Г.Фатеев

Пенза 2016

РЕФЕРАТ

Отчет содержит 100 с., 47 рис., 8 табл., 13 источников, 7 прил.

ЦЕЛОСТНОСТЬ ИНФОРМАЦИИ, АППАРАТУРА ПЕРЕДАЧИ ДАННЫХ, КОДЕР, ПОЛИНОМ, ПОМЕХОУСТОЙЧИВОСТЬ

Объектом исследований в дипломном проекте являются узлы обеспечения целостности данных.

Цель дипломного проекта — разработка узлов обеспечения целостности данных, используемых в аппаратуре передачи данных специального назначения и анализ влияния данных узлов на помехоустойчивость аппаратуры передачи данных специального назначения.

В процессе дипломного проектирования был проведен анализ факторов, влияющих на нарушение целостности данных; разработаны узлы целостности данных для факторов, влияющих на нарушение целостности, выявленных в процессе анализа; проведены исследования помехоустойчивости аппаратуры передачи данных на испытательном стенде в условиях воздействия гармонической помехи, накладываемой на исходный сигнал, при наличии узлов обеспечения целостности данных и при их отсутствии в аппаратуре передачи данных.

В результате дипломного проектирования было разработано узлы целостности данных и проведено исследование их эффективности.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное бюджетное образование учреждение
высшего образования

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УДК 004.056.2

Политехнический институт
Факультет приборостроения,
информационных технологий

УТВЕРЖДАЮ

Зав. кафедрой ИБСТ
к.т.н., доцент С.Л. Зефилов

и

электроники
Выпускающая кафедра:
Информационная
безопасность систем и технологий
Учебная группа 11 ПТ1

«___» _____ 2016 г.

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ
РАБОТУ**

студентки группы 11ПТ1 Подгорецкой Ольги Михайловны

Специальность 10.05.02 -

Информационная
безопасность
телекоммуникационных
систем

Тема дипломного проекта:

Узлы обеспечения целостности данных в аппаратуре передачи
данных специального назначения

Руководитель ВКР

инженер – электроник 2
категории АО «НПП «Рубин»

А.В. Бессонов

Нормоконтролер

к.т.н., доцент кафедры ИБСТ

А.П. Иванов

1 Цели и задачи ВКР

Цель работы – разработка узлов обеспечения целостности данных, используемых в аппаратуре передачи данных специального назначения и анализ влияния данных узлов на помехоустойчивость аппаратуры передачи данных специального назначения.

Узлы предназначены для обеспечения помехоустойчивости данных при передаче/приеме сообщений в аппаратуре передачи данных специального назначения.

2 Тактико–технические требования к выполнению ВКР

Технические требования:

- должен быть проведен анализ факторов, влияющих на нарушение целостности данных;
- должны быть разработаны узлы целостности данных для факторов, влияющих на нарушение целостности, выявленных в процессе анализа;
- должен быть разработан кодер на основе циклического кода (69,53) с обнаружением ошибок на основе образующего полинома $x^{16}+x^{12}+x^5+1$;
- при разработке узлов должно применяться средство автоматизированного проектирования Quartus II версии 8.1;
- должны быть проведены исследования помехоустойчивости аппаратуры передачи данных;
- исследования помехоустойчивости аппаратуры передачи данных должны быть проведены на испытательном стенде в условиях воздействия гармонической помехи, накладываемой на исходный сигнал, при наличии узлов обеспечения целостности данных и при их отсутствии в аппаратуре передачи данных;
- исследования помехоустойчивости должны быть проведены для:
 - скоростей передачи данных 1200, 2400 и 4800 бит/с;
 - уровней передаваемого сигнала 0,36 В, 0,8 В, 2,2 В;
- результаты проведенных исследований помехоустойчивости оформить в виде графиков зависимости коэффициентов ошибочного приема (BER) от соотношения сигнал/шум (помеха) (SNR) для всех видов проведенных испытаний.

В ВКР должны быть рассмотрены следующие вопросы:

- принципы построения аппаратуры передачи данных;
- проведение исследования помехоустойчивости аппаратуры передачи данных без использования узлов обеспечения целостности данных;
- анализ факторов, влияющих на нарушение целостности данных в аппаратуре передачи данных;
- разработка узлов обеспечения целостности данных для факторов, влияющих на нарушение целостности, выявленных в процессе анализа;
- проведение исследования помехоустойчивости аппаратуры передачи данных с использованием узлов обеспечения целостности данных.

3 Требования к разрабатываемой документации

Отчет по ВКР выполняется в соответствии с требованиями ГОСТ 7.32.

Перечень и содержание графической части ВКР:

- структура аппаратуры передачи данных – 1л., ф. А1 (плакат);
- структура узла «антидребезг» – 1л., ф. А1 (плакат);
- структура узла самоконтроля – 1л., ф. А1 (плакат);
- схема кодирующего устройства для циклического кода – 1л., ф. А1 (плакат);
- результаты экспериментальных исследований помехоустойчивости аппаратуры передачи данных без узлов обеспечения целостности данных – 1л., ф. А1 (плакат);
- результаты экспериментальных исследований помехоустойчивости аппаратуры передачи данных с использованием узлов обеспечения целостности данных – 1л., ф. А1 (плакат).

Электронная копия отчета и графической части ВКР должна быть представлена на магнитном или оптическом носителе.

4 Сроки выполнения ВКР

Тема утверждена приказом ректора ПГУ № ____ от «__» _____ 2016г.

Дата выдачи задания «__» _____ 2016г.

Время дипломного проектирования с 10 октября 2016г. по 15 января 2017г.

Задание к исполнению принял «__» _____ 2016г.

Исполнитель ВКР _____ Подгорецкая О.М.

СОДЕРЖАНИЕ

Введение.....	8
1 Принцип построения аппаратуры передачи данных.....	10
2 Анализ факторов влияющих на нарушение целостности данных в аппаратуре передачи данных	20
2.1 Нарушение целостности информации, вызванное физической неидеальностью элементов коммутации АПД.....	23
2.2 Нарушение целостности информации, вызванное искажением информации преднамеренными и непреднамеренными помехами в канале связи.....	25
2.3 Нарушение целостности информации, вызванное выходом из строя ЭРИ из состава АПД.....	33
3 Исследование помехоустойчивости аппаратуры передачи данных	43
3.1 Расчет необходимого объема статистического эксперимента	49
3.2 Экспериментальные исследования помехоустойчивости.....	52
3.2.1 Исследование помехоустойчивость АПД при заданном SNR.....	54
3.2.2 Исследование помехоустойчивости АПД при заданном уровне шума.....	64
Заключение	79
Список использованных источников	80
Приложение А. Код программы, реализующей защиту от дребезга	82
Приложение Б. Код программы, реализующей имитатор стыка RS-232	83
Приложение В. Код программы, реализующей кодер.....	86
Приложение Г. Код программы, реализующей декодер.....	88
Приложение Д. Код программы, реализующей адресацию (SK_1).....	92
Приложение Е. Код программы, реализующей проверку правильности принятия сообщения	98
Приложение Ж. Протокол проверки ВКР на оригинальность.....	100

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

АПД – аппаратура передачи данных

АСУ – автоматизированная система управления

КК – коробка контроля

ООД – оконечное оборудование данных

ПЛИС – программируемая логическая интегральная схема

ПО – программное обеспечение

ПРД – передатчик

ПРМ – приемник

$R_{\text{шума}}$ – уровень шума

ФЗ – федеральный закон

ЦВС – центральная вычислительная система

ЭВМ – электронно-вычислительная машина

ЭРИ – электро-радио изделие

BER – коэффициент ошибок на кодограмму

SNR – соотношение сигнал/шум

ВВЕДЕНИЕ

Национальная безопасность Российской Федерации существенным образом зависит от обеспечения информационной безопасности, и в ходе технического прогресса эта зависимость будет возрастать.

Под информационной безопасностью Российской Федерации понимается состояние защищенности её национальных интересов в информационной сфере, определяющих совокупность сбалансированных интересов личности, общества и государства [1].

Для обеспечения защищенности национальных интересов необходима современная армия, мобильная, оснащенная новым, надежным вооружением. А для того, чтобы это вооружение применялось эффективно, необходимо четкое, хорошо организованное взаимодействие между различными родами войск, оперативное управление боевыми машинами и комплексами в режиме реального времени. Эффективность взаимодействия во многом зависит от качества каналов связи и АПД, обеспечивающих достоверность передачи информации в условиях воздействия различных помех.

Поэтому исследование помехоустойчивости, анализ факторов влияющих на АПД и разработка узлов, обеспечивающих защиту данных, представляет большой интерес.

До 2015 года техника, поставляемая в Вооруженные Силы Российской Федерации, изготавливалась с использованием импортных ЭРИ, что делало ее зависимой от отношения иностранных государств к России, от санкций в отношении России. В настоящее время в стране проводится политика импортозамещения, в связи с этим изделия, поставляемые в вооруженные силы Российской Федерации, должны быть в максимально возможной степени выполнены на отечественной элементной базе.

Цель работы – разработка на отечественной элементной базе узлов обеспечения целостности данных, используемых в аппаратуре передачи

данных специального назначения и анализ влияния данных узлов на помехоустойчивость аппаратуры передачи данных.

Дипломный проект посвящен разработке узлов, предназначенных для обеспечения помехоустойчивости данных при передаче/приеме сообщений в аппаратуре передачи данных специального назначения. Для достижения поставленной цели в дипломном проекте рассматривается принцип построения АПД, проводится анализ факторов, влияющих на нарушение целостности данных, разрабатываются узлы обеспечения целостности данных для факторов, влияющих на нарушение целостности, выявленных в процессе анализа и проводится исследование помехоустойчивости аппаратуры передачи данных без и с использованием узлов обеспечения целостности данных.

1 Принцип построения аппаратуры передачи данных

Проведя анализ современных военных конфликтов, Российские военные специалисты сделали выводы, что требуется от современных АСУ комплексов противовоздушной обороны:

- высокая мобильность;
- простота эксплуатации;
- управление разнотипными комплексами;
- максимальная скорость работы средств вычисления;
- оперативность работы всего комплекса;
- возможность выполнения разнообразных задач [2].

Аппаратура передачи данных является неотъемлемой частью АСУ, а следовательно должна удовлетворять этим требованиям.

АПД представляет собой совокупность устройства преобразования сигналов, обеспечивающих прямое и обратное преобразование сигналов в вид, пригодный для передачи по каналу связи, устройства защиты от ошибок, входного кодера и выходного декодера канала передачи данных.

Данные - информация, представленная в виде, пригодном для обработки автоматическими средствами при возможном участии человека [3].

Информация – сведения (сообщения, данные) независимо от формы их представления [4].

Блок - схема общей системы передачи информации изображена на рисунке 1.

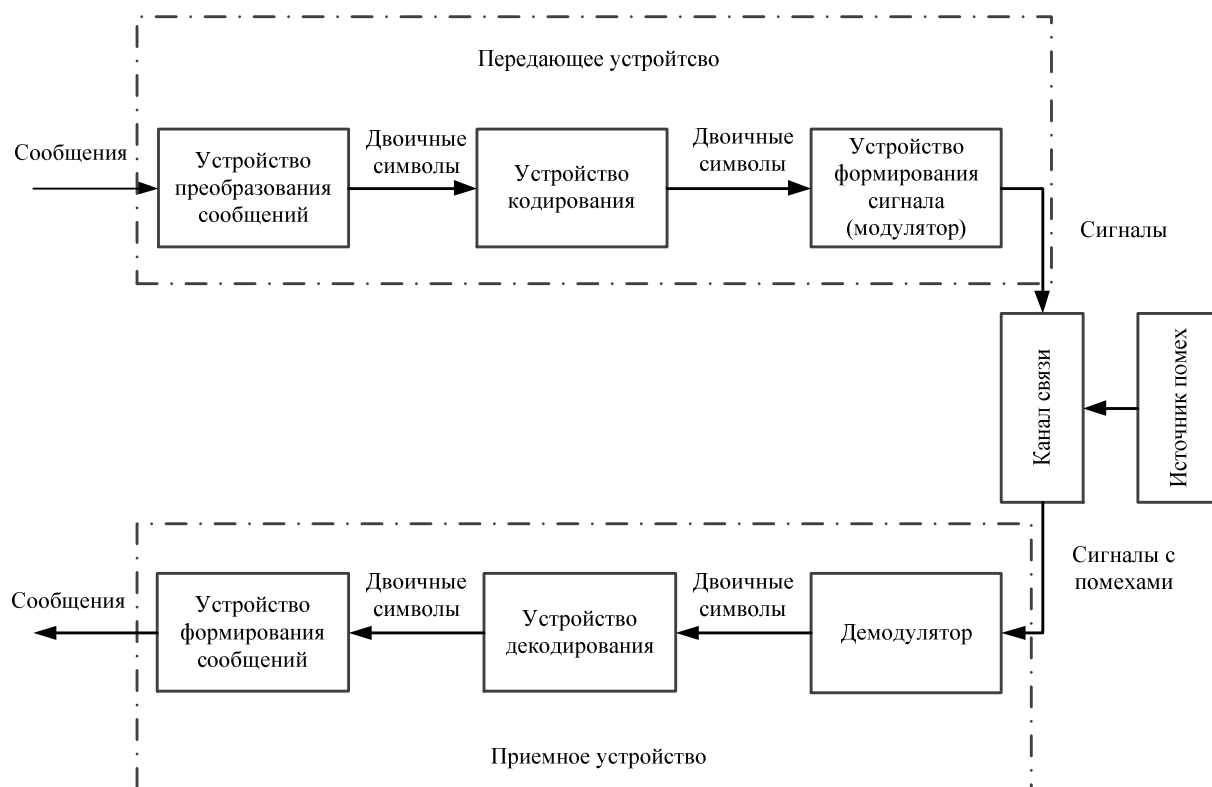


Рисунок 1 – Блок – схема общей системы передачи информации

Устройство преобразования сообщений служит для преобразования сообщений в вид, удобный для последующего кодирования и формирования сигнала. Если сообщение является непрерывным, а канал передачи данных дискретный, устройство преобразования сообщений осуществляет дискретизацию сообщения.

Устройство кодирования предназначается для ввода избыточности в передаваемую последовательность с целью её использования на приемной стороне для обнаружения и (или) исправления ошибок, возникших в процессе передачи по каналу связи.

Функцией модулятора является согласование сообщения источника или кодера со свойствами канала связи и обеспечение возможности одновременной передачи большого числа сообщений по одному каналу связи.

Канал связи представляет собой совокупность среды распространения сообщений и технических средств, обеспечивающих передачи сообщения от источника к получателю, которые разнесены в пространстве.

На приемной стороне происходят обратные преобразования: демодулятор преобразует принятые сигналы с помехами в последовательность двоичных символов, декодер обнаруживает и (или) исправляет ошибки, содержащиеся в последовательности двоичных символов, а устройство формирования сообщений преобразует последовательность в форму исходного сообщения.

Основными задачами АПД является прием, хранение и передача информации по радио и проводным каналам связи. Также АПД обеспечивает выполнение нижеследующих задач:

- хранение управляющей информации (адреса размещения данных обмена, набора управляющих признаков, определяющих режимы обмена);
- определение режима обмена;
- формирование, передачу и прием служебных сообщений (начальная синхронизация, пауза и т.п.).

В связи с задачами, которые выполняет АПД, не возникает вопросов об актуальности использования её для АСУ специального назначения, так как она позволяет в режиме реального времени обеспечивать доведение информации, необходимой для автоматизированного управления различными боевыми машинами и комплексами.

АПД, рассматриваемая в данном проекте, используется в качестве аппаратуры телекодowego приема и передачи данных центральной вычислительной системы боевой машины. Информация, получаемая АПД от пункта управления, в ЦВС используется для автоматизации процессов управления боевой машиной.

АПД, представляет собой приемопередатчик, предназначенный для работы по незащищенным импульсным радиоканалам и проводным каналам передачи данных.

Информация, которая циркулирует в АПД, актуальна в течение всего нескольких секунд и постоянно меняется. АПД обеспечивает передачу и прием дискретной информации реального масштаба времени при работе с абонентами со скоростями передачи (приема) данных: 1200 бит/с, 2400 бит/с, 4800 бит/с.

По проводным каналам, образованным телефонным кабелем, обеспечивается работа в технологическом (отладочном) режиме на расстояние до 100 м.

АПД представляет собой одноплатный модуль, выполненный в конструктиве «Евромеханика» на базе многослойной печатной платы с размерами (160 x 233) мм² (формат 6U). Модуль устанавливается в ЭВМ ЦВС в качестве дополнительного модуля и занимает в ней одно установочное место (рисунок 2).

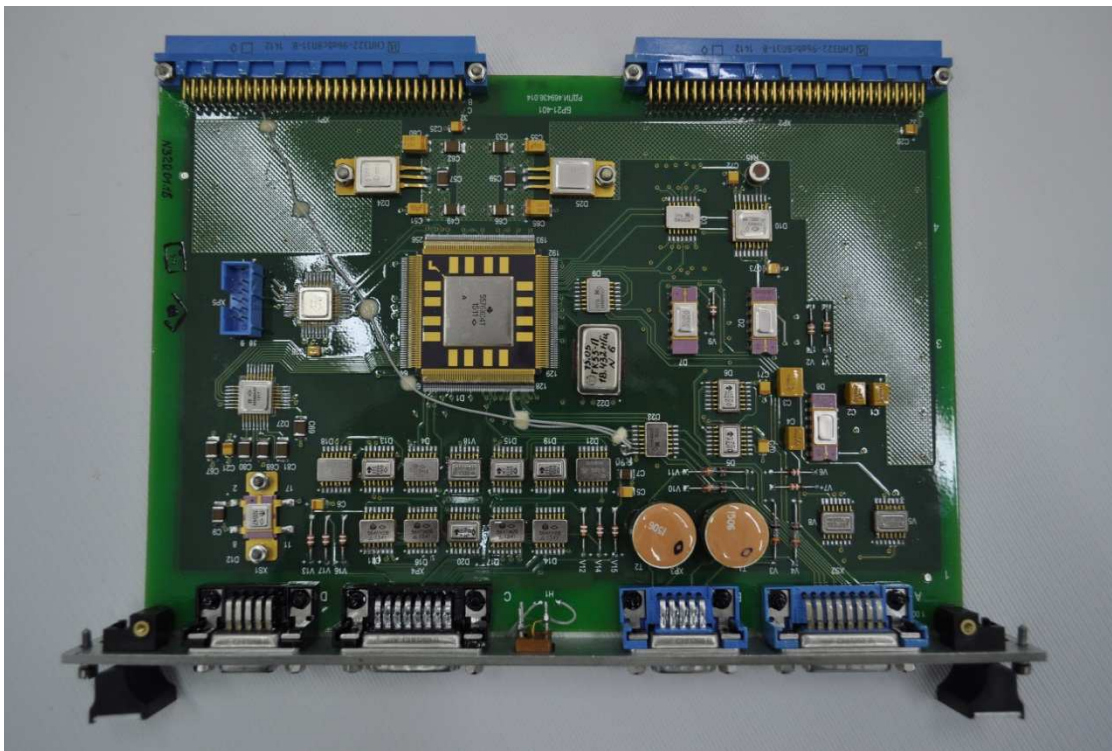


Рисунок 2 – Внешний вид модуля

Все основные цифровые узлы АПД реализованы на ПЛИС. Проектирование ПЛИС выполнялось в системе автоматизированного проектирования (САПР) Quartus II, версия 8.1.

Разработку узлов ПЛИС в САПР Quartus II можно выполнять тремя способами:

1 с помощью графических элементов. Данный способ реализации узлов возможен в случае если проект состоит из небольшого количества графических элементов и связей между ними, иначе проект теряет свою разборчивость. Для создания проекта используются существующие в САПР Quartus II элементы.

Примером такой реализации случит генератора псевдослучайной последовательности на ПЛИС фирмы Altera [5] показанные на рисунке 3.

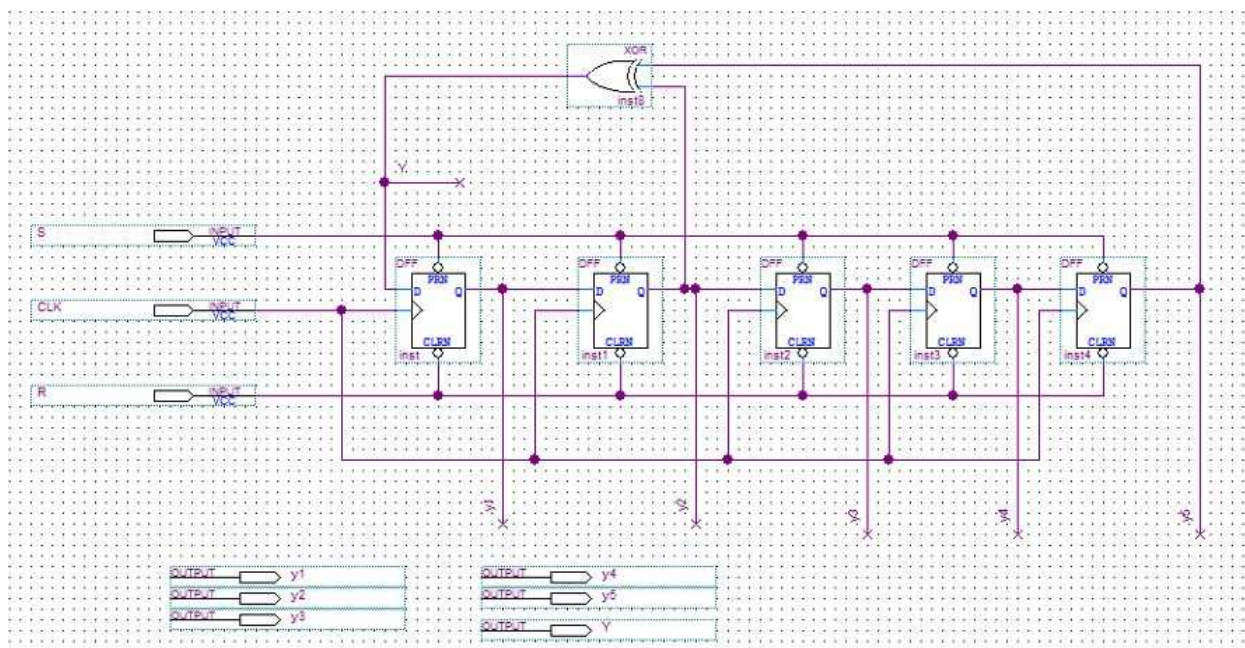


Рисунок 3 – Пример графической реализации узла

2 с помощью языка Very high speed integrated circuits Hardware Description Language, сокращенно VHDL. Одним из наиболее полезных свойств VHDL является возможность применения так называемого «функционального» описания, когда разработчик не конкретизирует структуру схемы, а только описывает её поведение, то есть то, как она должна работать. Составление самой электрической схемы возлагается на программу – компилятор, которая визуализирует код в виде одного графического элемента (рисунок 4). При двойном нажатии на графический объект «coder1» откроется код объекта.

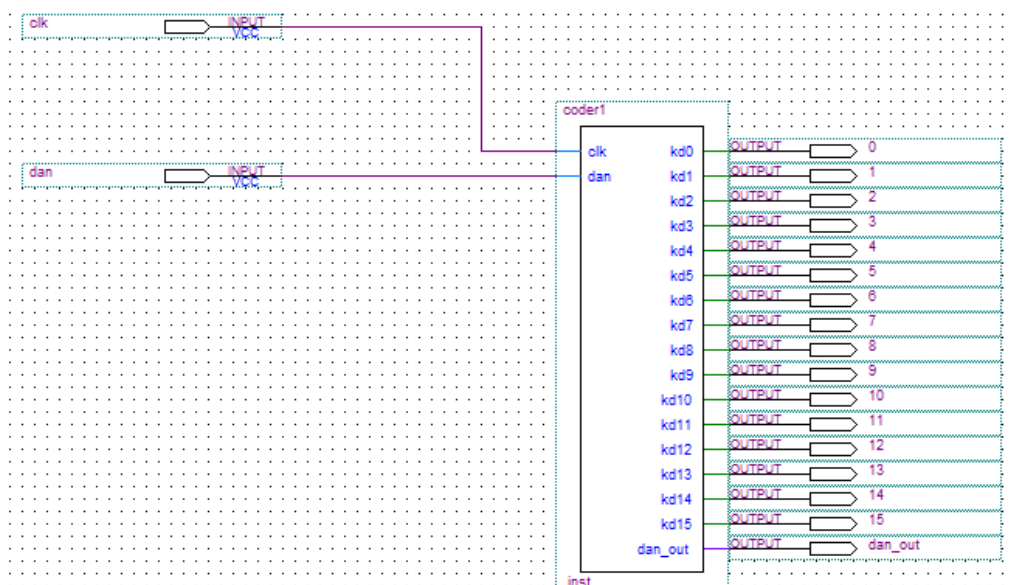


Рисунок 4 – Пример визуализации

Преимуществом реализации проекта на языке VHDL является:

- скорость проектирования: с помощью VHDL проще и быстрее ввести и проверить большой проект;
- проект на VHDL это объединение структуры устройства и алгоритма его функционирования;
- универсальность проекта: разработанный однажды проект может быть перенесен в другой проект;
- долговечность проекта: разработанный однажды проект может быть повторно использован через несколько лет на другой элементной базе;
- надежность проекта: синтаксический анализ, моделирование и компиляция в логическую схему быстро выявляют ошибки проекта.

3 с помощью языка VHDL и графических элементов. Данный способ реализации узлов наиболее предпочтителен из-за совмещения достоинств каждого из способов.

Пример такой реализации показан на рисунке 5.

Здесь узлы IMM_UART2 и UART_1 написаны на языке VHDL, а узел room_imm – графический элемент, взятый из библиотеки элементов.

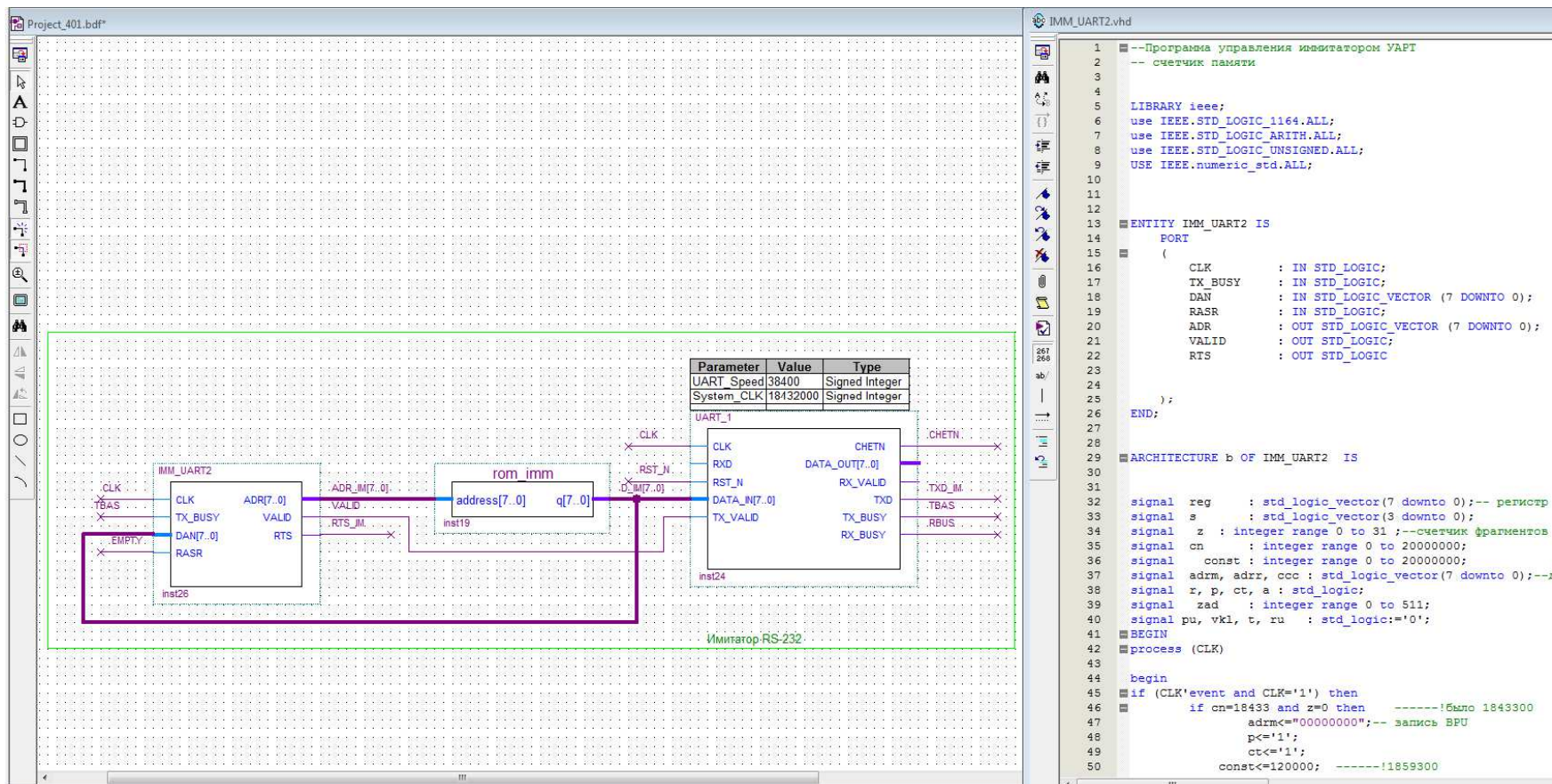


Рисунок 5 – Пример реализации узла с помощью языка VHDL и графических элементов

Синтаксис и семантика языка VHDL, на котором написаны программы, определяется стандартом ГОСТ Р 50754-95 [6].

В данной АПД используются отечественные ЭРИ, входящие в состав перечня электронной компонентной базы, разрешенной для применения при разработке, модернизации, производстве и эксплуатации вооружения, военной и специальной техники.

В качестве ПЛИС используется интегральная микросхема 5576XC4T ОАО «КТЦ «Электроника», г. Воронеж, обладающая возможностью многократного изменения конфигурации в составе аппаратуры. Микросхема предназначена для использования в цифровых интегральных модулях средств связи, предназначенных для управления бортовым комплексом радиосвязи летательных аппаратов, приема, обработки и передачи речевой и телекодированной информации [7].

Благодаря своим габаритам (ширина 43,4мм и вес не более 21г), объему встроенной памяти (96Кбит), количеству логических элементов (9984), типовой логической емкости (200000 вент.) и количеству выводов, программируемых пользователем (171), ПЛИС позволила увеличить производительности системы, снизить массу конечного изделия и уменьшить его габариты.

Программирование загрузочного ПЗУ модуля (микросхемы 5576PC1U) производится в составе модуля с помощью ЭВМ с программным обеспечением САПР Quartus II, версия 8.1, с использованием кабеля USB-Blaster Download Cable, подключаемого к USB-порту ЭВМ с одной стороны и разъему интерфейса JTAG программируемого модуля с другой. Конфигурационный файл прошивки ПЗУ с расширением .prof создается автоматически программным обеспечением САПР Quartus II, версия 8.1 при проектировании ПЛИС.

Микросхема 5576PC1U является перепрограммируемым ПЗУ, поэтому во время наладки модуля можно многократно менять данные

конфигурации ПЛИС, что облегчает процесс отладки функциональных узлов ПЛИС.

При включении питания модуля данные конфигурации ПЛИС загружаются в ПЛИС из ПЗУ.

2 Анализ факторов влияющих на нарушение целостности данных в аппаратуре передачи данных

Для АПД специального назначения остро стоит вопрос обеспечения безопасности информации, так как АПД специального назначения предназначена для охраны безопасности государства. Рассмотрим безопасность информации на примере трех основных свойств безопасности: конфиденциальность, целостность и доступность.

Согласно № 149 – ФЗ, конфиденциальность информации – обязательное для выполнения лицом, получившим доступ к определенной информации, требование не передавать такую информацию третьим лицам без согласия её обладателя [4]. Конфиденциальности могут угрожать:

- перехват информации техническими средствами разведки;
- навязывание ложной информации.

Перехват (информации) – неправомерное получения информации с использованием технических средств, осуществляющего обнаружение, прием и обработку информационных сигналов.

Нарушение конфиденциальности не повлечет за собой серьёзных последствий. Перехваченная информация теряет свою актуальность через 10с, поэтому перехват информации не оказывает влияния на процессы управления боевой машиной.

Навязывание ложной информации также не оказывают серьезного влияния на работу ЦВС, так как реализованные в ней алгоритмы, позволяют определить примерное положение цели и в случае, если полученные координаты сильно отличаются от определенных системой, полученные данные фильтруются (отсев ложных целей).

Согласно № 149 – ФЗ, доступ к информации – возможность получения информации и её использования [4]. Доступности данных в АПД могут угрожать:

- отказ АПД или её узлов;

- выход из строя ЭРИ из состава АПД.

Доступность информации является важным свойством для любой АПД, а для АПД специального назначения этот вопрос стоит достаточно остро. Для обеспечения доступности информации необходим периодический контроль состояния АПД, заключающийся в проверке работоспособности ЭРИ и составных узлов АПД. Данный контроль позволяет не только определить исправность/неисправность АПД, но и установить место, в котором произошел сбой; данное свойство контроля позволяет существенно сократить время на нахождение неисправности. Контроль за состоянием АПД можно производить в двух вариантах:

- самоконтроль;
- человеческий контроль.

Самоконтроль выполняется при каждом включении АПД и периодически во время работы АПД без участия человека. Смысл самоконтроля заключается в подаче определенного набора данных с передатчика на приемник и анализе полученных приемником данных.

Человеческий контроль заключается в периодическом техническом обслуживании АПД и проверке ее работоспособности согласно методике проверки. Эта методика включает в себя выполнение определенного набора тестов.

Понятие целостности информации определяется как состояние информации, при котором отсутствует любое её изменение, либо изменение осуществляется только преднамеренно субъектами, имеющими на него право [8]. Целостности данных в АПД могут угрожать:

- физическая неидеальность элементов коммутации АПД;
- искажения информации, вызванные преднамеренными и непреднамеренными помехами в канале связи;
- выход из строя ЭРИ из состава АПД.

Контроль целостности данных чрезвычайно важен при передаче команд управления, а также при хранении информации о поступивших командах и меняющихся параметрах. Нарушение целостности данных в АПД влечет за собой проблемы взаимодействия между источником информации и получателем. Под проблемами взаимодействия понимаем неполный прием данных, прием некорректных (измененных) данных, отказ в приеме данных.

Для обеспечения целостности данных могут применяться следующие способы:

- использование элементов задержки;
- контроль целостности сообщения по четности и нечетности;
- помехоустойчивое кодирование;
- самоконтроль.

Некоторые из этих способов защиты можно реализовать как программным способом, так и аппаратным. Каждый из двух способов защиты имеет свои достоинства и недостатки. Так, например, при аппаратной реализации элемента задержки, в схему достаточно поставить один диод, который будет срабатывать с задержкой, достаточной для исключения возможности попадания в канал связи импульсов, вызванных переключением элементов коммутации. При реализации этого же способа защиты программно, необходимо использовать элемент с памятью, который можно запрограммировать. В данном случае наиболее простым и удобным способом является использование диода, но так как в электрической принципиальной схеме на АПД предполагается использование ПЛИС, то на её основе можно программным способом реализовать элемент задержки и не ставить дополнительный элемент в схему. В случае помехоустойчивого кодирования реализация кодера программно значительно проще, быстрее и экономичнее (как по денежным затратам, так и по размерам получившегося изделия) по сравнению с аппаратной реализацией.

Если рассматривать такие способы защиты как контроль целостности сообщения по четности и нечетности и самоконтроль, то их реализация аппаратными способами не целесообразна, так как реализация этих способов защиты займет очень много места на плате.

Таким образом каждый для данной АПД из способов защиты от нарушения целостности данных реализуется программным способом на ПЛИС.

2.1 Нарушение целостности информации, вызванное физической неидеальностью элементов коммутации АПД

Физическая неидеальность элементов коммутации АПД (например, переключателя выбора канала передачи данных) обусловлена тем, что ключи элементов коммутации имеют упругость, их коммутация – сложный процесс. После первого соударения контактов при коммутации происходит ряд упругих отскоков, называемым дребезгом контактов, поэтому вместо однократного изменения логического сигнала возникает серия импульсов, способная создать множество ложных переключений. Например, для переключателя поворотного типа ПГ2 время установления однозначного состояния составляет 550 мс. Данный процесс показан на рисунке 6.

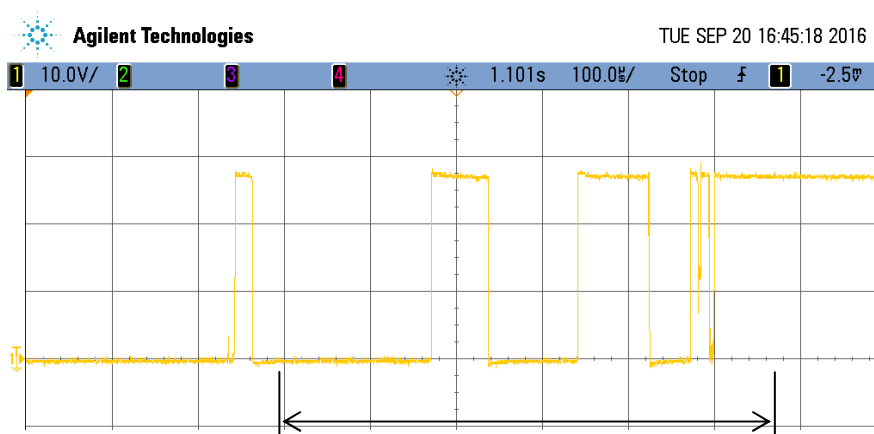


Рисунок 6 – Время переключения переключателя

Для получения очищенного от «дребезга» сигнала используется простейший программный элемент задержки, реализованный программно на языке VHDL ПЛИС. Схема работы узла «антидребезг» приведена на рисунке 7.



Рисунок 7 – Схема работы узла «антидребезг»

Данный узел, изменяет значения выходного сигнала в случае, если идет участок нестабильности входного сигнала, что указывает на процесс изменения положения механического элемента или значение входного сигнала четко установлено (рисунок 8).

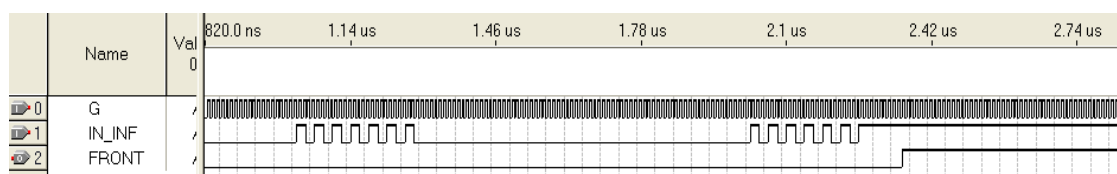


Рисунок 8 - Изменения выходного сигнала

Если после периода, когда значение входного сигнала определено однозначно, идет интервал «дребезга», что говорит об изменении положения

механического элемента, узел воспринимает этот дребезг как сброс выходного сигнала (рисунок 9).

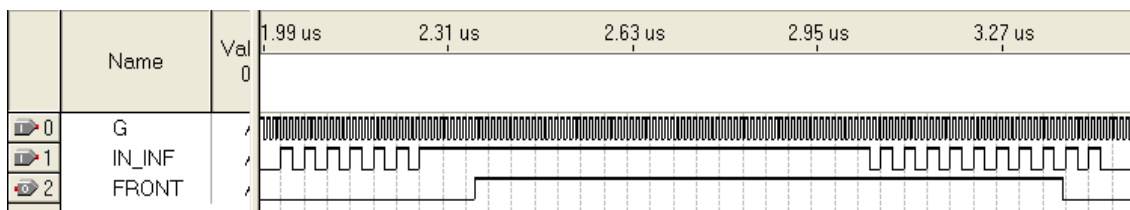


Рисунок 9 - Сброс сигнала

При переключении механического элемента введение программной задержки позволяет избежать как потери данных в конце сообщения, так и «отрезать» не принадлежащий данному сообщению предшествующий элемент.

Код программы приведен в приложении А.

2.2 Нарушение целостности информации, вызванное искажением информации преднамеренными и непреднамеренными помехами в канале связи

При воздействии на сигнал в канале связи наведенных (преднамеренных) и непреднамеренных («белый» шум) помех, на приеме после демодуляции сообщения может быть искажено, следовательно его целостность нарушится. Для того, чтобы избавиться от ошибок при приеме информации или отбросить сообщения, с нарушенной целостностью, могут использоваться:

- контроль целостности сообщения по четности и нечетности;
- помехоустойчивое кодирование.

Контроль целостности по четности/нечетности предусматривает сложение всех бит сообщения по модулю 2. Результат становится контрольным битом. Контроль четности по контрольной сумме осуществляется путем подсчета контрольной суммы сообщения. В

простейшем примере контрольная сумма сообщения формируются путем арифметического сложения байтов без учета бита переноса.

Рассмотрим пример реализации проверки сообщения на нечетность при реализации контроллера интерфейса RS-232, интерфейс RS-232 предназначен для подключения аппаратуры, передающей или принимающей данные (ООД, или АПД). Так в передающей части интерфейса на месте девятого бита сообщения формируется бит проверки на нечетность путем сложения по модулю 2 всех 8 информационных бит:

```
chet<=DATA_IN(0) XOR DATA_IN(1) XOR DATA_IN(2) XOR
DATA_IN(3) XOR DATA_IN(4) XOR DATA_IN(5) XOR DATA_IN(6) XOR
DATA_IN(7);
when bit = 9 =>
    TXD <= chet;
    TxBitCnt <= TxBitCnt+1;
    CHETN<=chet
```

А в приемной части осуществляется проверка данного бита суммированием всех 8 принятых информационных бит и бита нечетности. Если результат проверки равен 1, то осуществляется сброс сигнала готовности к приему и сообщение отбрасывается.

```
when 9 =>
    if (CHETN XOR RxBuf(0) XOR RxBuf(1) XOR RxBuf(2) XOR
RxBuf(3) XOR RxBuf(4) XOR RxBuf(5) XOR RxBuf(6) XOR RxBuf(7)) = '1'
then
    RxReady <= '0';
    Else
    RxReady <= '1';
    end if;
```

Полный код программы, реализующий имитатор стыка RS-232 написанный в соответствии с ГОСТ Р 50754-95 [10], приведен в приложении Б.

Помехоустойчивость – понятие, характеризующее способность системы передачи информации противостоять искажающему действию помехи [9].

Кодирование сообщений осуществляется с помощью Кодера. Задачей данного узла АПД является введение избыточности в передаваемое сообщение с целью его защиты от ошибок, возникающих при воздействии помех. Для этого информационная часть сообщения делится на образующий полином, полученный остаток от деления определяет последовательность проверочных элементов. Кодер построен на основе циклического кода с обнаружением ошибок. Обнаружение ошибок является по своей природе задачей намного более простой, чем исправление ошибок. Так как аппаратура передачи данных имеет двусторонний канал, легче обеспечить обнаружение ошибок и запросить повторную передачу сообщения, чем пользоваться кодом, исправляющим ошибки, тратя на это время и ресурсы. Циклический код это разновидность симметрических кодов, характеризующийся тем, что циклическая перестановка всех символов одной комбинации дает другую комбинацию, принадлежащую этому же коду. В качестве многочлена используется полином $x^{16}+x^{12}+x^5+1$. Полином – это формально заданный степенной ряд, т.е. сумма множества степенных выражений независимых переменных. Выбор образующего полинома продиктован требованиями ГОСТ 17422-82 [10]. Кодовая комбинация полинома 1 0001 0000 0010 0001 используется для обнаружения одиночных и двойных ошибок, а также для обнаружения групповых ошибок длиной не более 16 бит с вероятностью 99,997% [11]. Основные характеристики для кода, построенного на основе данного полинома:

- длина кода $n = 69$;
- число информационных разрядов $k = 53$;
- число контрольных разрядов $r = 16$;
- основание кода $m = 2$;
- общее число кодовых комбинаций $m^n = 2^{69}$;
- число разрешенных кодовых комбинаций $N_p = 2^{53}$.

Схема кодера систематического кода $(69,53)$ приведена на рисунке 10.

Здесь $a(x)$ – входной поток информации; $u(x)$ – выходной поток, ТИ – тактовый импульс.

Основой кодера так же является регистр сдвига с логическими связями, структура которых определяется порождающим многочленом. В схеме кодера систематического кода регистр сдвига содержит 16 разрядов, 3 сумматора по модулю 2 и 15 условных связей с умножителями. За первые 53 такта на вход кодера и одновременно на выход поступает информационное слово $a(x)$. В течение этого времени замыкается обратная связь через &1 и в регистре формируется остаток $r(x)$, элементы которого поступают на выход через схему &2 за последующие 16 тактов.

Длина регистра соответствует степени многочлена $g(x)=16$. Число логических связей сумматора по модулю 2 равно 3, так как четыре коэффициента порождающего многочлена за исключением старшего равны единице $g_0=g_5=g_{12}=g_{16}=1$. Размещаются сумматоры по модулю 2 перед ячейками, номера которых соответствуют индексам единичных коэффициентов.

Код программы, приведен в приложении В.

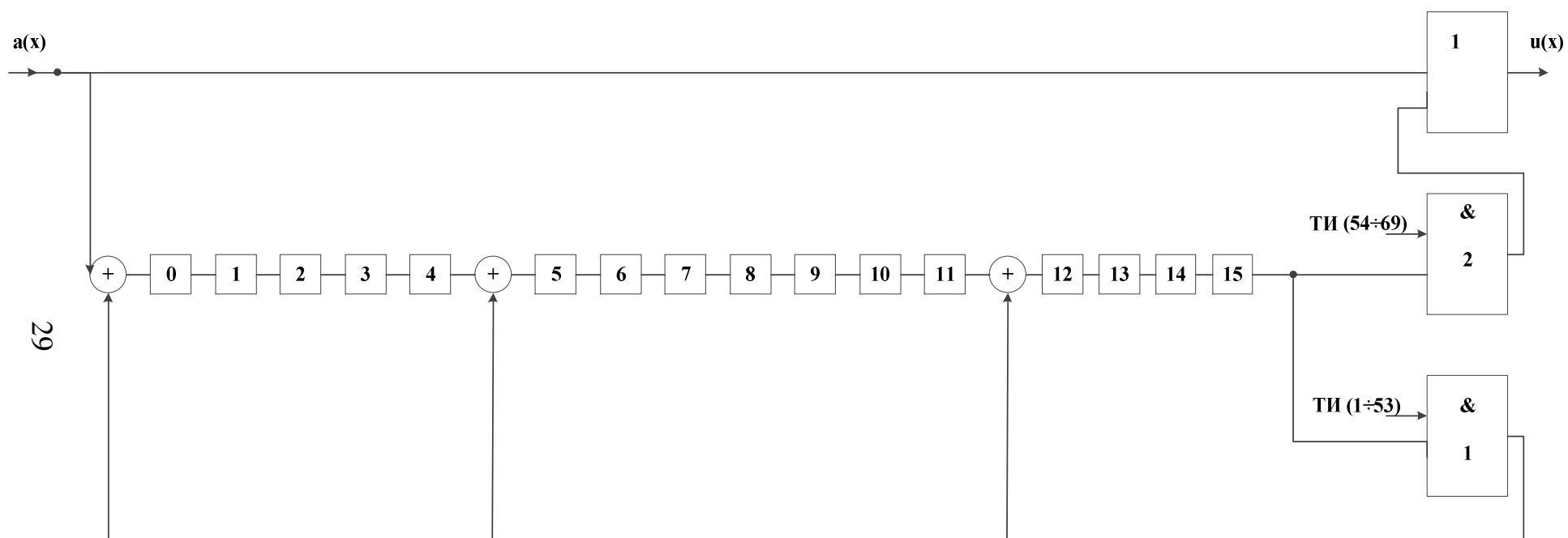


Рисунок 10 - Схема кодера систематического кода $(69, 53)$

В приведенной ниже таблице 1 показаны состояния ячеек в процессе работы регистра.

Таблица 1 – Информация на выходе сдвигающего регистра

		Номер разряда (выход)																
Номер такта	Вход	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Выход
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
8	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
9	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0
10	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0
11	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0
12	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0
13	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0
14	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0
15	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0
16	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0
17	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0
18	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
19	0	0	0	1	0	1	1	1	0	1	0	1	0	0	0	1	0	1
20	1	0	0	0	1	0	1	1	1	0	1	0	1	0	0	0	1	0
21	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0	0	1
22	1	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0	0
23	0	1	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0
24	1	0	1	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0
25	0	1	0	1	0	0	0	0	0	1	1	1	1	1	0	1	0	0
26	1	0	1	0	1	0	0	0	0	0	1	1	1	1	1	0	1	0
27	1	0	0	1	0	1	1	0	0	0	0	1	1	0	1	1	0	1
28	0	1	0	0	1	0	1	1	0	0	0	0	1	1	0	1	1	0
29	1	1	1	0	0	1	1	1	1	0	0	0	0	0	1	0	1	1
30	0	0	1	1	0	0	0	1	1	1	0	0	0	1	0	1	0	1
31	1	0	0	1	1	0	0	0	1	1	1	0	0	0	1	0	1	0
32	0	0	0	0	1	1	1	0	0	1	1	1	0	1	0	1	0	1
33	1	0	0	0	0	1	1	1	0	0	1	1	1	0	1	0	1	0
34	0	0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	0	1

Продолжение таблицы 1

Номер такта	Вход	Номер разряда (выход)																Выход
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
35	1	0	0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	0
36	0	0	0	0	0	0	1	0	0	1	1	0	0	0	1	0	0	1
37	1	0	0	0	0	0	0	1	0	0	1	1	0	0	0	1	0	0
38	0	1	0	0	0	0	0	0	1	0	0	1	1	0	0	0	1	0
39	1	1	1	0	0	0	1	0	0	1	0	0	1	0	0	0	0	1
40	0	1	1	1	0	0	0	1	0	0	1	0	0	1	0	0	0	0
41	1	0	1	1	1	0	0	0	1	0	0	1	0	0	1	0	0	0
42	0	1	0	1	1	1	0	0	0	1	0	0	1	0	0	1	0	0
43	1	0	1	0	1	1	1	0	0	0	1	0	0	1	0	0	1	0
44	0	0	0	1	0	1	0	1	0	0	0	1	0	1	1	0	0	1
45	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	1	0	0
46	0	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	1	0
47	1	1	1	0	0	0	0	0	1	0	1	0	0	1	1	0	1	1
48	0	0	1	1	0	0	1	0	0	1	0	1	0	1	1	1	0	1
49	1	0	0	1	1	0	0	1	0	0	1	0	1	0	1	1	1	0
50	0	0	0	0	1	1	1	0	1	0	0	1	0	0	0	1	1	1
51	0	1	0	0	0	1	0	1	0	1	0	0	1	1	0	0	1	1
52	1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	0	0	1
53	0	1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	0	0
54	0	0	1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	0
55	0	0	0	1	1	1	0	0	0	0	0	1	0	1	0	0	0	1
56	0	0	0	0	1	1	1	0	0	0	0	0	1	0	1	0	0	0
57	0	0	0	0	0	1	1	1	0	0	0	0	0	1	0	1	0	0
58	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1	0	1	0
59	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1	0	1
60	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1	0
61	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1
62	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
63	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0
64	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0
65	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0
66	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
68	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
69	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

структуру которого определяет выбранный образующий полином.

Декодирование проверочных и информационных разрядов отличается. Это отличие вызвано особенностями электрической схемы аппаратуры передачи данных, в частности наличием инвертора на входе ПЛИС, и требованиями ГОСТ 28082-89, из которого следует, что проверочные разряды представляют собой инверсию остатка от деления передаваемого блока данных на образующий полином по модулю 2 [11].

При декодировании последовательность, поступившая на вход декодера проверяется по методике, изложенной в приложении 1 ГОСТ 28082-89 и по значению полученного остатка $R^*(x)$ от деления принятого блока данных на образующий полином делается заключение о наличии или отсутствии ошибок в принятых данных.

Код программы, приведен в приложении Г.

2.3 Нарушение целостности информации, вызванное выходом из строя ЭРИ из состава АПД

Выход из строя ЭРИ одна из самых распространенных причин отказа любой аппаратуры. Причины выхода из строя ЭРИ весьма разнообразны: производственный брак ЭРИ, электрический пробой, климатические воздействия и другие. Отказ ЭРИ может произойти как во время работы АПД, так и во время простоя, поэтому проверку состояния АПД необходимо производить при каждом включении и периодический по время работы. Для проведения подобной проверки в АПД реализован режим самоконтроля. При самоконтроле вход приемника автоматически отключается от канала связи и замыкается на выход передатчика (организуется внутренний шлейф), необходимым условием является установка приемника и передатчика в одинаковые режимы (по скорости, виду работы). Полученное от ЭВМ в цикле «Запись передатчика» сообщение возвращается обратно в ЭВМ в цикле «Чтение приемника». В ЭВМ при приеме информации от приемника она сравнивается с переданной в передатчик, и по результатам сравнения

делается вывод об исправности приемопередатчика. Такая проверка должна проводиться на разных скоростях передачи и при разных видах работы.

В режиме самоконтроля модуль АПД формирует тестовые сообщения и выдает их на вход передатчика АПД по интерфейсу RS-232. Результат самоконтроля выдается в ЭВМ в слове состояния модуля. Для обеспечения возможности организации самоконтроля в модуль введен узел самоконтроля (имитатора интерфейса RS-232).

В модуле АПД использовались сигналы интерфейса RS-232, приведенные в таблице 2.

Таблица 2 – Используемые сигналы интерфейса

Сигнал	Назначение цепи
DTR	ЭВМ готова
DSR	АПД готова
RTS	Запрос передачи
CTS	Готов к передаче
TXD	Передаваемые данные
RXD	Принимаемые данные
GND	Сигнальное заземление

Асинхронный режим передачи является байт-ориентированным (символьно-ориентированным): минимальная пересылаемая единица информации — один байт (один символ). Формат асинхронной посылки позволяет выявлять возможные ошибки передачи. Формат посылки байта иллюстрирует рисунок 12.



Рисунок 12 - Формат посылки

Каждое сообщение, используется для передачи одной кодограммы или блока информации, содержащего режимы работы модуля или состояние модуля.

Все сообщения состоят из заголовка сообщения и информационной части сообщения, содержащей данные. Формат сообщения приведен в таблице 3.

Таблица 3 – Формат сообщения

Формат сообщения	Поля сообщения	Длина поля
Заголовок сообщения	Идентификатор заголовка	2 байта
	Идентификатор сообщения	1 байт
Информационная часть сообщения	Данные: - кодограмма; - блок режимов работы модуля; - регистр состояния модуля.	Определяется типом операции в идентификаторе сообщения

Идентификатор заголовка – двухбайтовая последовательность (коды F0 hex и 75 hex), используются как указатель начала сообщения.

Идентификатор сообщения определяет код операции и количество информационных байтов. Идентификаторы сообщения никогда не совпадают с идентификатором заголовка.

Код операции определяет направление передачи информации (запись/чтение) и условный номер абонента.

Если принят перепад, сигнализирующий о начале посылки, а по стробу старт-бита зафиксирован уровень логической единицы, старт-бит считается ложным и приемник снова переходит в состояние ожидания. Об этой ошибке приемник может не сообщать.

Если во время, отведенное под стоп-бит, обнаружен уровень логического нуля, фиксируется ошибка стоп-бита.

Если применяется контроль четности, то после посылки бит данных передается контрольный бит. Этот бит дополняет количество единичных бит данных до четного или нечетного в зависимости от принятого соглашения. Прием байта с неверным значением контрольного бита приводит к фиксации ошибки.

Контроль формата позволяет обнаруживать обрыв линии: как правило, при обрыве приемник “видит” логический нуль, который сначала трактуется как старт-бит и нулевые биты данных, но потом срабатывает контроль стоп-бита.

В АПД используется асинхронный старт-стопный режим со следующими основными параметрами:

- скорость передачи – 115200 бит/с;
- число информационных бит в посылке – 8;
- число стоповых бит – 1;
- с проверкой на нечетность;
- данные передаются, начиная с младшего бита.

Схема взаимодействия ЭВМ и АПД приведена на рисунке 13.

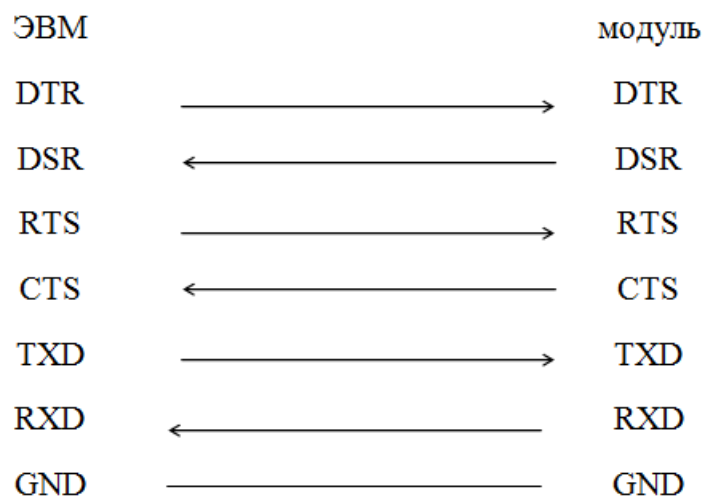


Рисунок 13 - Схема связей ЭВМ и АПД

Обмен информацией между ЭВМ и АПД организуется в соответствии с временными диаграммами, приведенными на рисунках 14, 15.

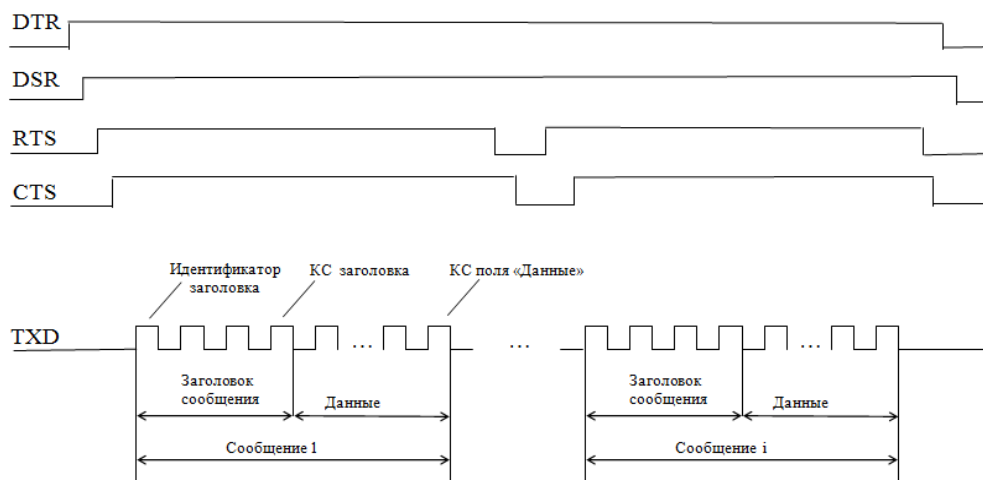


Рисунок 14 - Временная диаграмма обмена информацией между ЭВМ и модулем при операциях «Запись ПРД», «Запись блока режимов работы модуля»

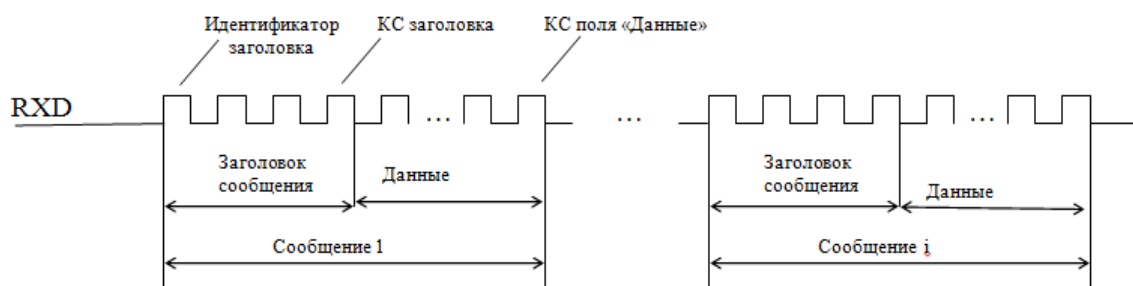


Рисунок 15 - Временная диаграмма обмена информацией между ЭВМ и модулем при операциях «Чтение ПРМ», «Чтение блока режимов работы модуля»

ЭВМ в случае готовности к взаимодействию устанавливает сигнал DTR. Модуль формирует сигнал готовности и устанавливает сигнал DSR (готовность АПД).

Обмен сообщениями между ЭВМ и АПД выполняется при наличии активного состояния цепей DTR и DSR.

Для передачи любого сообщения в модуль ЭВМ активизирует сигнал на линии RTS (запрос на передачу). Модуль выставляет в ответ сигнал CTS, после чего ЭВМ по цепи TXD выдает сообщение, содержащее блок режимов работы модуля или кодограмму для передачи в канал связи и снимает сигнал RTS. Получив полностью сообщение от ЭВМ, после снятия сигнала RTS модуль снимает цепь CTS.

Цикл «Запись блока режимов работы модуля» организуется ЭВМ при начальной загрузке модуля и при необходимости изменения режимов работы модуля.

После получения от ЭВМ блока режимов работы модуль организует операцию «Чтение блока режимов работы модуля» и возвращает полученную информацию в ЭВМ с целью контроля, для чего по цепи RXD выдает сообщение, содержащее заданные режимы работы.

Передача информационных сообщений в модуль (цикл «Запись ПРД») организуется ЭВМ при наличии данных на передачу в АПД и при наличии свободного места в буфере ПРД. Информация о наличии свободного места в буфере ПРД выдается в ЭВМ в слове состояния модуля.

Передача в ЭВМ служебного сообщения с пакетом состояния модуля производится по обновлению состояния, но не чаще, чем раз в 5 мсек. Если обновления нет, передача пакета с информацией состояния происходит периодически, один раз в сек.

Кодограммы, предназначенные для выдачи в канал связи, записываются в буфер передатчика. Размер буфера – 4 кодограммы. Передатчик (ПРД) устанавливает в регистре состояния признак занятости этого буфера и фиксирует количество байтов кодограмм в нем в слове состояния модуля после получения каждого информационного сообщения от ЭВМ. Содержимое регистра состояния изменяется также при чтении очередной кодограммы из буфера ПРД для передачи в канал связи.

Выдача кодограмм в канал связи осуществляется в том порядке, в котором они были записаны в буфер ПРД модуля (по принципу FIFO – первый вошел, первый вышел).

Цикл «Чтение ПРМ» организуется модулем при наличии в приемнике информации, полученной из канала связи, независимо по отношению к циклам передачи сообщений от ЭВМ в модуль.

При передаче информации в ЭВМ от модуля циклы «Чтение блока режимов работы модуля» и «Чтение регистра состояния» имеют более высокий приоритет по сравнению с циклами «Чтение ПРМ».

Если при приеме от ЭВМ сообщения нарушен формат хотя бы одной посылки (нет бита «стоп», обнаружена ошибка при проверке контрольной суммы), модуль снимает цепь CTS и принятое сообщение считается недействительным. Для возобновления передачи ЭВМ должна снять цепь RTS в ответ на снятие цепи CTS и затем через время $t \geq 2$ мс вновь ее

установить для организации нового цикла передачи сообщения.

Если количество информационных байтов не соответствует заданному в идентификаторе принимаемого сообщения, сообщение также считается недействительным.

В любом из вышеперечисленных случаев нарушения формата передачи принятая кодограмма не выдается в канал связи, в блоке режимов работы сохраняется предыдущая информация.

При обращении ЭВМ с операцией «Запись ПРД» к выключенному передатчику в регистре состояния устанавливается соответствующий признак ошибки, и модуль организует цикл «Чтение регистра состояния модуля».

При включении электропитания в модуле автоматически осуществляется самоконтроль. В режиме самоконтроля модуль АПД формирует тестовые сообщения и выдает их на вход передатчика АПД по интерфейсу RS-232. Результат самоконтроля выдается в ЭВМ в слове состояния модуля. Для обеспечения возможности организации самоконтроля в модуль введен узел самоконтроля (имитатора интерфейса RS-232).

На рисунке 16 изображены составные части узла самоконтроля (имитатора интерфейса RS-232).

Узел самоконтроля состоит из следующих узлов:

- SK_1 – формирователь адреса памяти имитатора;
- SK_MEM – память данных имитатора;
- UART_2 – формирователь сигналов интерфейса RS-232 (RTS_IM, TXD-IM);
- проверка правильность сообщений.

Код программы, реализующей формирование адреса памяти имитатора, приведен в приложении Д.

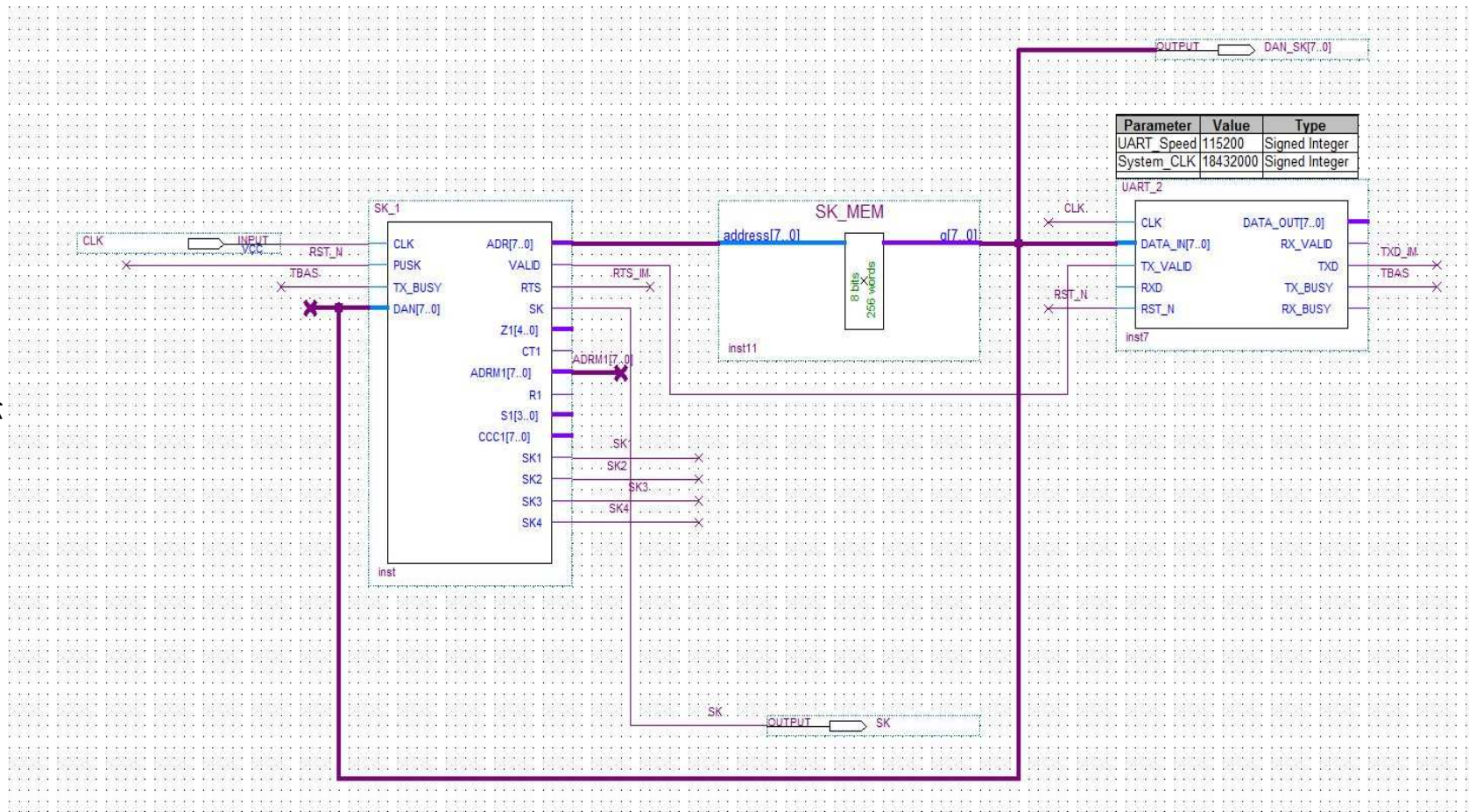


Рисунок 16 - Узел самоконтроля

Узел проверки правильности принятия сообщения находится отдельно от узла самоконтроля. Данный узел располагается в приемнике. Код программы, реализующий проверку правильности принятия сообщения, приведен в приложении Е.

3 Исследование помехоустойчивости аппаратуры передачи данных

При передаче данных по каналу связи, на передающийся информационный сигнал накладываются помехи.

Помеха – это любое воздействие, накладывающееся на полезный сигнал и затрудняющее его прием. Помехи весьма разнообразны как по своему происхождению, так и по физическим свойствам [12].

В каналах связи основной вид помех - импульсные шумы и прерывание связи. Появление импульсных помех часто связано с автоматической коммутацией. Прерывание связи это явление, при котором сигнал в линии либо совсем исчезает, либо резко затухает.

Также, в любой аппаратуре при любом диапазоне частот имеют место внутренние шумы. Внутренние шумы это хаотическое движение носителей заряда в усилительных приборах, сопротивлениях и других элементах аппаратуры.

Классифицировать помехи можно следующим признакам:

- по происхождению;
- по характеру воздействия на сигнал;
- по физическим свойствам.

К помехам по происхождению относятся:

- внутренние шумы аппаратуры;
- помехи от посторонних источников, находящихся вне каналов связи:
 - промышленные помехи, возникающие в электрических цепях электроустановок;
 - помехи от посторонних станций и каналов, возникающих от различных нарушений режима их работы и свойств каналов;
 - атмосферные помехи (грозовые разряды и др.), обусловленные электрическими процессами в атмосфере [12].

По характеру воздействия на сигнал различают:

- аддитивные помехи;
- мультипликативные помехи.

Мультипликативная помеха – это помеха, мгновенные значения которой перемножаются с мгновенными значениями сигнала. Мешающее действие таких помех проявляется в виде изменения параметров полезного сигнала, в основном амплитуды. В реальных каналах электросвязи обычно имеют место не одна, а совокупность помех.

Аддитивной называется помеха, мгновенные значения которой складываются с мгновенными значениями сигнала. Мешающее воздействие аддитивной помехи определяется суммированием с полезным сигналом. Аддитивные помехи воздействует на приемное устройство независимо от сигнала и имеют место даже тогда, когда на входе приемника отсутствует сигнал [12].

Аддитивные помехи разделяются на три вида:

- гармонические;
- импульсные;
- флуктуационные.

Гармоническими называют помехи, спектр которых сосредоточен в узкой полосе частот, такие помехи могут вызываться источниками питания, другими радиостанциями и т.д. Спектр сигнала – распределение энергии сигнала по частотам.

Импульсные помехи – это случайные последовательности относительно коротких импульсов, создаваемые промышленными установками и атмосферными источниками.

Флуктуационные помехи – непрерывные во времени случайные процессы с нормальным распределением амплитуд и нулевым средним значением. Наличие этих помех вызвано радиоизлучением звезд, луны, радиационными поясами земли.

Исследование воздействия аддитивных помех на АПД является неотъемлемой частью создания АПД, так как наличие данных помех в канале связи влияет на качество полученных данных (аддитивные помехи, складываясь на информативный сигнал, нарушают его целостность). Данные исследования проводятся на испытательном стенде. Наибольший интерес, для разработчиков, представляет аддитивные гармонические помехи, так как они могут находиться на той же частоте, что и частота, на которой происходит обмен данными.

По физическим свойствам помехи делятся на два вида:

- флуктуационные помехи;
- сосредоточенные помехи.

Флуктуационные помехи - случайный процесс с нормальным распределением (гауссов процесс). Этот вид помех практически имеет место во всех реальных каналах.

Электрическая структура флуктуационной помехи это последовательность бесконечно коротких импульсов, имеющих случайную амплитуду и следующих друг за другом через случайные промежутки времени. Эти импульсы появляются настолько часто, что переходные явления в приемнике от отдельных импульсов накладываются, образуя случайный непрерывный процесс.

Сосредоточенные помехи делятся на два вида:

- сосредоточенные по времени;
- сосредоточенные по спектру.

Сосредоточенные по времени помехи - помехи в виде одиночных импульсов, следующих один за другим через большие промежутки времени, поэтому переходные явления от одного импульса успевают практически затухнуть к моменту прихода следующего импульса [12].

К сосредоточенным по спектру помехам принято относить сигналы посторонней АПД, излучения генераторов высокой частоты различного

назначения и т. п. В отличие от флуктуационных и импульсных помех, спектр которых заполняет полосу частот приёмника, ширина спектра сосредоточенной помехи меньше полосы пропускания приёмника. В диапазоне коротких волн этот вид помех является основным, определяющим помехоустойчивость связи [12].

Исследование помехоустойчивости узлов обеспечения целостности данных АПД проводится на испытательном стенде, который включает в себя следующие компоненты:

- источник питания;
- преобразователь напряжения;
- переходное устройство;
- модуль;
- ЭВМ;
- генератор гармонической помехи;
- коробка контроля;
- специальное программное обеспечение.

В качестве источника питания, имитирующего напряжение бортовой сети используется источник питания постоянного тока Б5-71/1мм, на котором выставлено напряжение 27В.

Преобразователь напряжения предназначен для преобразования напряжения 27В в напряжения питания модуля – 12В, минус 12В и 5В, преобразователь напряжения входит в состав переходного устройства, которое предназначено для подключения модуля и подачи на модуль напряжения питания.

В качестве ЭВМ выступает персональный компьютер, на который установлено специальное программное обеспечение (ПО), написанное для тестирования данного модуля.

Генератор шума – устройство, имитирующее помеху в канале связи.

Коробка контроля предназначена для сложения чистого сигнала с шумом.

Специальное ПО – программа, формирующая управляющие и информационные сообщения для тестирования модуля АПД, принимающая от АПД информационные сообщения и сообщения о состоянии модуля, анализирующая их и выдающая на экран монитора сообщения о результатах контроля.

Пример работы ПО показан на рисунке 17.

```
f0 75 19 80 88 44 22 11 88 83
f0 75 19 80 88 44 22 11 88 83 W

В ТЕСТЕ N5a В РЕЖИМЕ N1 :

ПРОВЕРКА ВЫПОЛНЕНА.
ВИД РАБОТЫ ВР: ОК_ОДК2
ВИД КАНАЛА ВК: РАДИОКАНАЛ
УРОВЕНЬ ПЕРЕДАЧИ УПД: 0.8 V
ТИП РАДИОСТАНЦИИ РС: "АРБАЛЕТ"
СКОРОСТЬ ПЕРЕДАЧИ СПД: 1200 бод
ПАКЕТНАЯ ПЕРЕДАЧА: 1 КДГР

ОШИБКА ПРИ СРАВНЕНИИ ИНФОРМАЦИИ.

1156 - кол-во выданных в модуль кдгр
----- ИЗ НИХ-----
1112 - кол-во принятых из модуля кдгр
43 раз(а) - пропадание кдгр - вообще нет !

Для продолжения нажмите клавишу <ENTER>!
```

Рисунок 17 – Пример работы ПО

Схема испытательного стенда для исследования помехоустойчивости АПД в условиях воздействия гармонической помехи, накладываемой на исходный сигнал показана на рисунке 18.

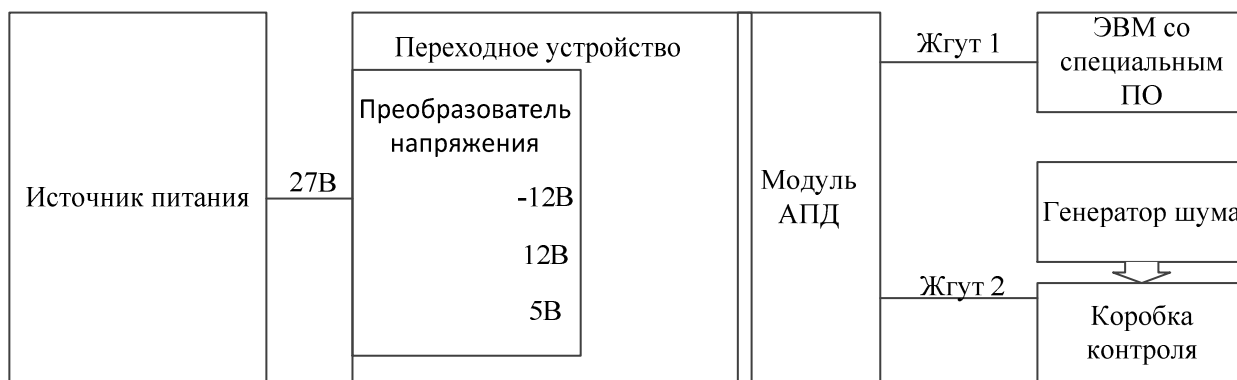


Рисунок 18 - Схема испытательного стенда для исследования помехоустойчивости АПД

Схема, показанная на рисунке 18 работает следующим образом: модуль АПД устанавливается в переходное устройство. Электропитание модуля осуществляется от преобразователя напряжения, размещаемого в переходном устройстве. Модуль подключается жгутом 1 к ЭВМ со специальным ПО, жгутом 2 – к коробке контроля.

От ЭВМ в передатчик модуля поступают проверочные кодограммы по интерфейсу RS-232, где преобразуются в аналоговый сигнал для передачи в канал связи. Из передатчика модуля аналоговый сигнал поступает в коробку контроля, где складывается с гармоническими помехами. Из коробки контроля результат сложения информативного сигнала и помехи поступает на приемник модуля АПД, преобразуется в цифровой вид и выводится на ЭВМ, где ПО обрабатывает полученное сообщение, определяет наличие искажений в нем и считает количество правильно и неправильно принятых сообщений. Описанная схема взаимодействия показана на рисунке 19.

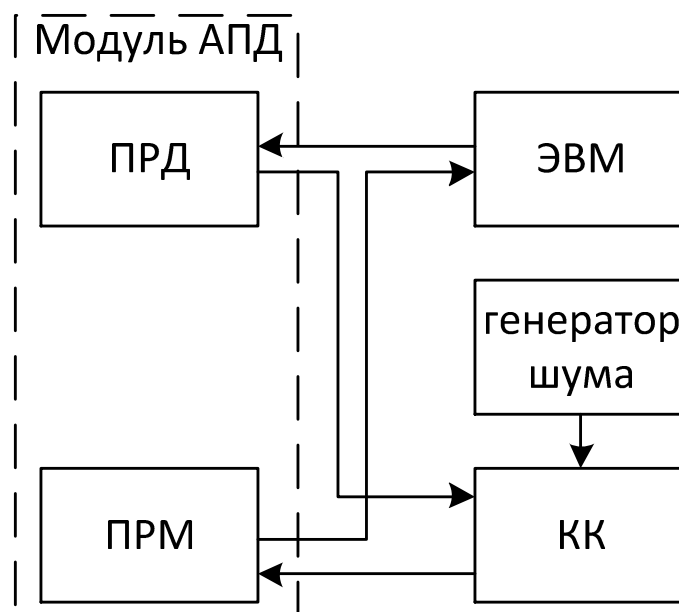


Рисунок 19 - Схема взаимодействия модуля с ЭВМ и КК

Исследования проводятся на следующих скоростях передачи данных: 1200, 2400, 4800бит/с, при уровне передаваемого сигнала 0,36В, 0,8В, 2,2В.

3.1 Расчет необходимого объема статистического эксперимента

Точность экспериментальной оценки вероятности ошибки зависит от количества испытаний. Будем рассматривать передачу каждой кодограммы как одно независимое вероятностное испытание с двумя возможными исходами:

- правильный прием;
- ошибочный прием.

В этом случае необходимый объем испытаний можно вычислить пользуясь интегральной теоремой Муавра-Лапласа, согласно которой справедливо уравнение (1).

$$P_d \left(\left| \frac{m}{n} - p \right| < \varepsilon \right) \approx \Phi \left(\varepsilon \sqrt{\frac{n}{pq}} \right) \quad (1)$$

где m – число произошедших событий (число ошибок);

n – общее число испытаний (объем статистического эксперимента);

p – вероятность того, что событие произойдет (вероятность ошибки);

q – вероятность того, что событие не произойдет (вероятность правильного приема);

ε – требуемая точность оценки вероятности (доверительный интервал);

$\Phi(x)$ – функция Крампа, определяемая выражением (2)

$$\Phi(x) = \sqrt{\frac{2}{\pi}} \int_0^x e^{-\frac{t^2}{2}} dt \quad (2)$$

$P_D \left(\left| \frac{m}{n} - p \right| < \varepsilon \right)$ – вероятность того, что отношение будет с оценкой p с абсолютной погрешностью меньше ε (доверительная вероятность).

Интеграл вероятности определяется по формуле (3)

$$F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\frac{x-\mu}{\sqrt{D}}} e^{-\frac{t^2}{2}} dt \quad (3)$$

Функция Крампа связана с интегралом вероятностей соотношением

$$\Phi(x) = 2 \left[F(x) - \frac{1}{2} \right] = 2F(x) - 1 \quad (4)$$

С учетом (4) выражение (1) принимает вид:

$$P_d \left(\left| \frac{m}{n} - p \right| < \varepsilon \right) = 2F \left(\varepsilon \sqrt{\frac{n}{pq}} \right) - 1 \quad (5)$$

откуда

$$F \left(\varepsilon \sqrt{\frac{n}{pq}} \right) = \frac{1}{2} \left[P_d \left(\left| \frac{m}{n} - p \right| < \varepsilon \right) + 1 \right] \quad (6)$$

Задаваясь значениями доверительного интервала ξ и доверительной вероятности $P_d \left(\left| \frac{m}{n} - p \right| < \varepsilon \right)$, при каждом значении p на основе соотношения (6) можно найти необходимое значение n . Для этого:

1) по заданной величине P_d рассчитывается численное значение y , равное правой части выражения (6)

$$y = \frac{1}{2} \left[P_d \left(\left| \frac{m}{n} - p \right| < \varepsilon \right) + 1 \right]$$

2) определяется значение $x = \varepsilon \sqrt{\frac{n}{pq}}$ аргумента функции $F \left(\varepsilon \sqrt{\frac{n}{pq}} \right)$, при котором выполняется равенство (6).

3) из уравнения $x = \varepsilon \sqrt{\frac{n}{pq}}$ определяется необходимое значение объема статистического эксперимента.

$$n = \frac{x^2}{\varepsilon^2} pq = \frac{x^2}{\varepsilon^2} p(1 - p) \quad (7)$$

Для проведения исследования помехоустойчивости доверительная вероятность (P_d) примем равной 90% или 0,9; требуемая точность оценки вероятности (ε) равна или 0,2р; вероятность ошибки (p) также равна 5% или 0,05. Исходя из приведенной выше методики расчета необходимого объема статистического эксперимента, получается:

$$y = \frac{1}{2} \left[P_d \left(\left| \frac{m}{n} - p \right| < \varepsilon \right) + 1 \right] = 0,95;$$

$$x = \varepsilon \sqrt{\frac{n}{pq}} = 1,6448536$$

$$n = \frac{x^2}{\varepsilon^2} pq = \frac{x^2}{\varepsilon^2} p(1 - p) = 12850$$

Число экспериментов, которые необходимо провести для получения 90% вероятности правильного приема данных, составляет 12850 раз.

Один эксперимент составляет 100 циклов, таким образом, общее число циклов, равно 1285000. Один цикл - 10 кодограммы, общее число отправленных и принятых кодограмм равно 12850000.

3.2 Экспериментальные исследования помехоустойчивости

Оценка помехоустойчивости производится на основе рассмотрения соотношения между сигналом и помехой, при котором обеспечивается заданная величина коэффициента ошибок на бит.

Отношение сигнал/шум (SNR) – это отношение полезного сигнала и нежелательного шума, которое находится по формуле (8).

$$SNR = \frac{P_{\text{сигнала}}}{P_{\text{шума}}} \quad (8)$$

Привычнее видеть соотношение сигнал/шум, выраженное в логарифмических единицах с использованием децибел (9).

$$SNR_{dB} = 10 \log_{10} \left(\frac{P_{\text{сигнала}}}{P_{\text{шума}}} \right) \quad (9)$$

Коэффициент ошибок по битам (BER – Bit error ratio)– отношение числа принятых с ошибками бит к числу посланных бит, вычисляемое за определенный период времени.

Величина коэффициента колеблется около значения среднего коэффициента ошибок за длительный промежуток времени. Разница между непосредственно измеренным коэффициентом ошибок и долговременным средним значением зависит от числа контролируемых бит и тем самым от длительности измерения. База времени формируется при помощи двух основных методов:

— на принимающей стороне задается число наблюдаемых бит регистрируется число бит с ошибками. Достоинством данного метода является точно известное время измерения, недостатком - невысокая точность при малом числе измерений;

— на принимающей стороне задается число бит с ошибками, измерение длится до тех пока не будет зафиксировано заданное число ошибок. Недостаток заключается в том, что неизвестно время измерений, которое при малых коэффициентах ошибок может оказаться очень большим [13].

В данной работе используется первый метод определения коэффициента ошибок по битам.

Так как используемое ПО подсчитывает количество правильно и неправильно принятых кодограмм, то коэффициент BER будем считать коэффициентом ошибок по кодограммам. Другими словами, будем считать, что BER – это отношение числа принятых с ошибками кодограмм к числу посланных кодограмм, вычисляемое за определенный период времени.

Экспериментальные исследования помехозащищенности АПД проводились для скоростей передачи 1200, 2400 4800 бит/с при уровне передаваемого сигнала 0,36В, 0,8В и 2,2В без использования узлов обеспечения целостности и с их использованием. Экспериментальные исследования помехозащищенности АПД проводились в двух вариантах:

- исследование помехоустойчивость АПД при заданном SNR;
- исследование помехоустойчивости АПД при заданном уровне шума.

Первое исследование проводится с целью проверки работоспособности АПД при разных уровнях сигнала и разных скоростях передачи данных, проверяется работоспособность узлов обеспечения целостности данных.

Второе исследование проводится с целью проверки эффективности работы узлов обеспечения целостности при разных уровнях сигнала, скорости и разном уровне шума.

3.2.1 Исследование помехоустойчивость АПД при заданном SNR

Результаты исследования помехоустойчивости АПД без узлов обеспечения целостности к влиянию аддитивного белого гауссова шума приведены в таблице 4.

Таблица 4- Результаты исследования помехоустойчивости АПД без узлов обеспечения целостности

	BER								
Скорость передачи	1200			2400			4800		
Уровень сигнала	0,36	0,8	2,2	0,36	0,8	2,2	0,36	0,8	2,2
SNR, дБ									
-1	0,482	0,483	0,482	0,5	0,5	0,5	0,5	0,5	0,5
0	0,48	0,477	0,481	0,5	0,5	0,5	0,5	0,5	0,5
1	0,479	0,478	0,479	0,5	0,5	0,5	0,5	0,5	0,5
2	0,475	0,461	0,47	0,5	0,5	0,5	0,5	0,5	0,5
3	0,445	0,442	0,44	0,5	0,5	0,5	0,5	0,5	0,5
4	0,359	0,354	0,349	0,5	0,496	0,489	0,5	0,5	0,5
5	0,319	0,329	0,326	0,447	0,461	0,457	0,5	0,5	0,5
6	0,269	0,279	0,254	0,377	0,391	0,356	0,5	0,5	0,497
7	0,212	0,199	0,2	0,297	0,279	0,28	0,416	0,39	0,392
8	0,174	0,164	0,168	0,244	0,23	0,235	0,341	0,322	0,329
9	0,155	0,156	0,164	0,217	0,219	0,23	0,304	0,306	0,322

По данным, приведенным в таблице 4 построенным графики зависимости коэффициента ошибок на бит BER от соотношения сигнал/шум SNR, приведенные на рисунках 20 - 25.

На рисунках 20-22 приведена зависимость BER от SNR при постоянной скорости передачи. Как видно из рисунков, изменение уровня сигнала незначительно влияет на значение коэффициента BER. Это связано с тем, что при проведении исследований задавалось значение отношения сигнал/шум, а не значение шума. Для разного уровня сигнала, при одном и том же значении коэффициента SNR, уровень шума разный.

На рисунках 23-25 приведена зависимость BER от SNR при постоянном уровне сигнала. Следует обратить внимание на наличие связи

между скоростью передачи информации и помехоустойчивостью. Увеличение скорости передачи данных (при фиксированной полосе частот канала и мощности передатчика) приводит к уменьшению помехоустойчивости. И наоборот, снижение скорости передачи информации (при фиксированной полосе частот канала и мощности передатчика) приводит к увеличению помехоустойчивости.

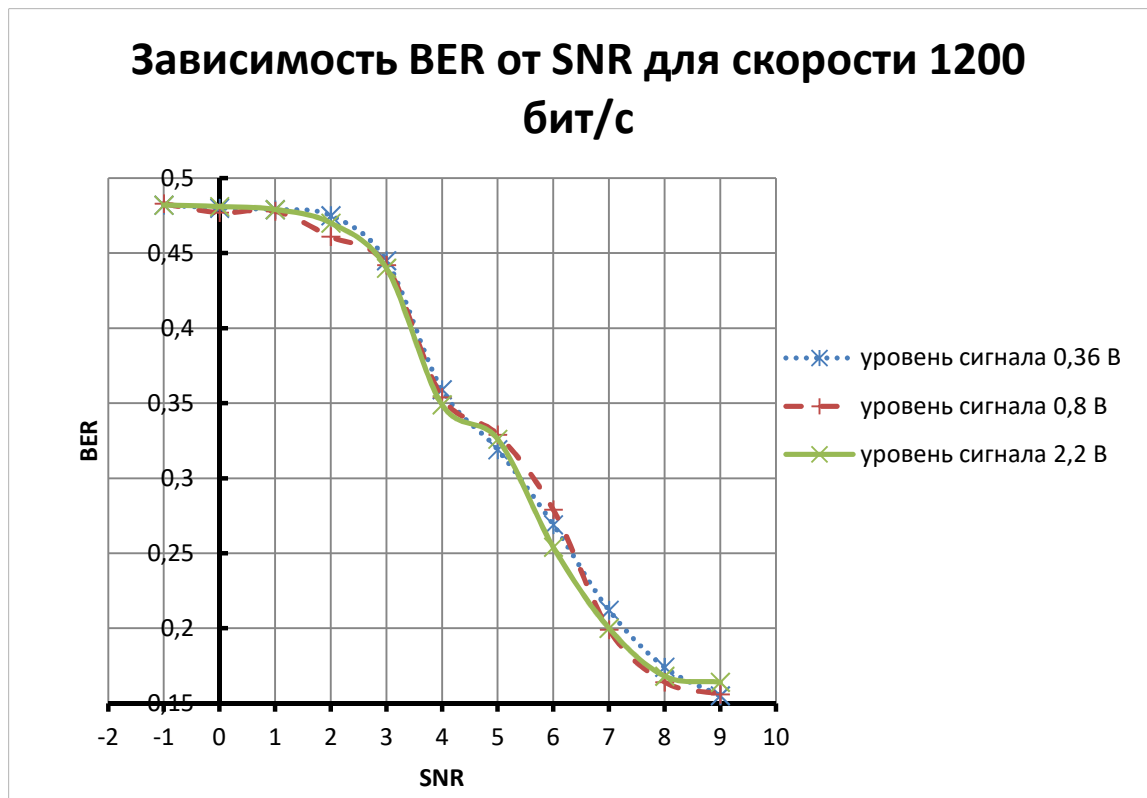


Рисунок 20 - Зависимость BER от SNR для скорости 1200 бит/с

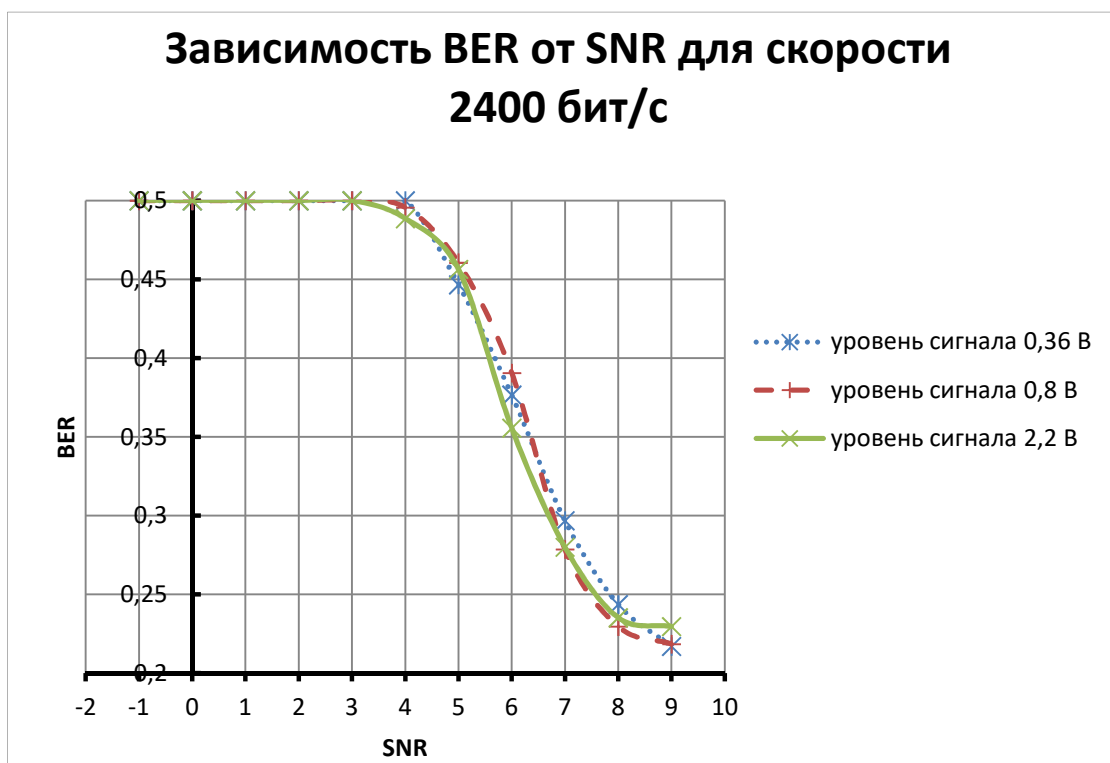


Рисунок 21 - Зависимость BER от SNR для скорости 2400 бит/с

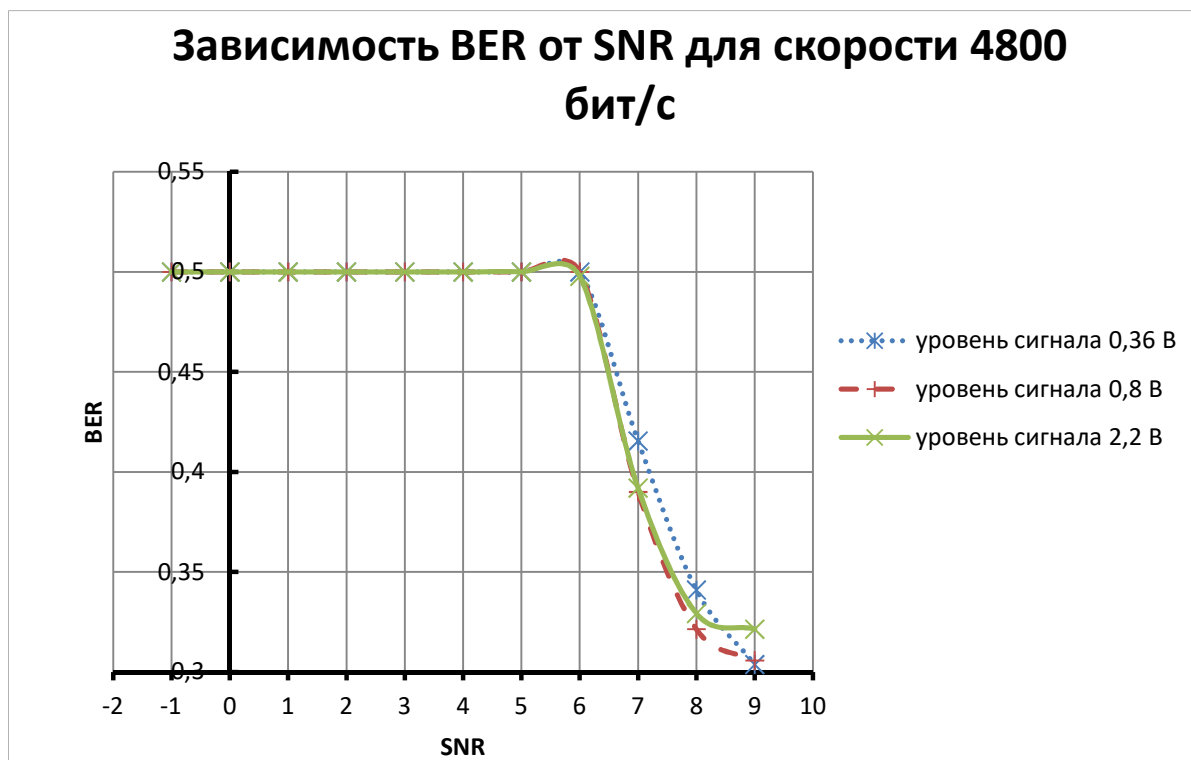


Рисунок 22 - Зависимость BER от SNR для скорости 4800 бит/с

Зависимость BER от SNR для сигнала уровнем 0,36 В

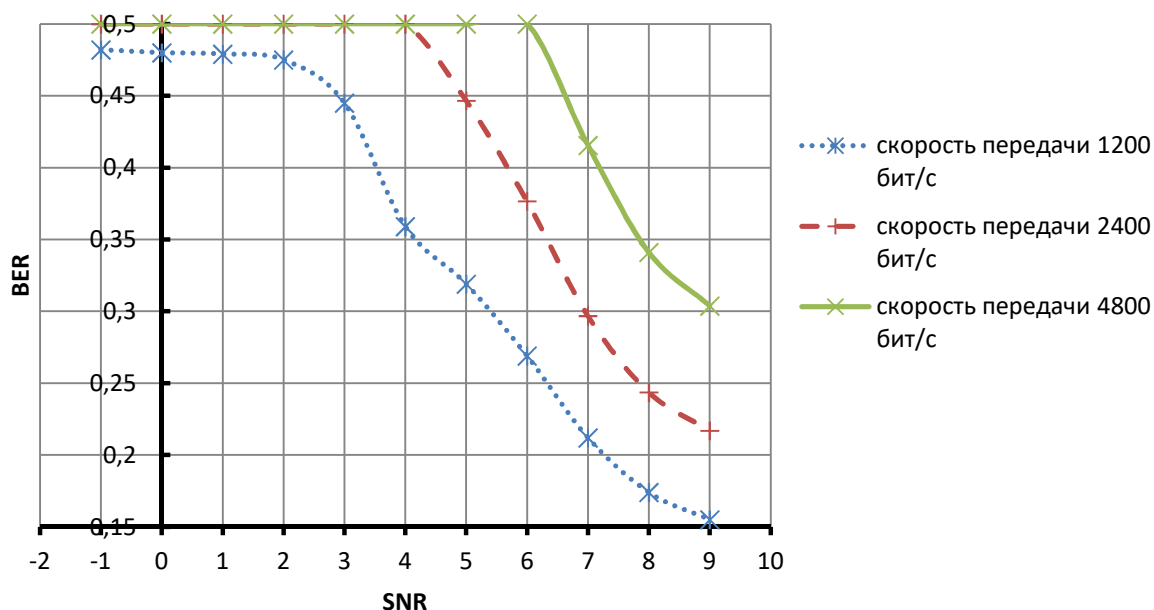


Рисунок 23 - Зависимость BER от SNR для сигнала уровнем 0,36 В

Зависимость BER от SNR для сигнала уровнем 0,8 В

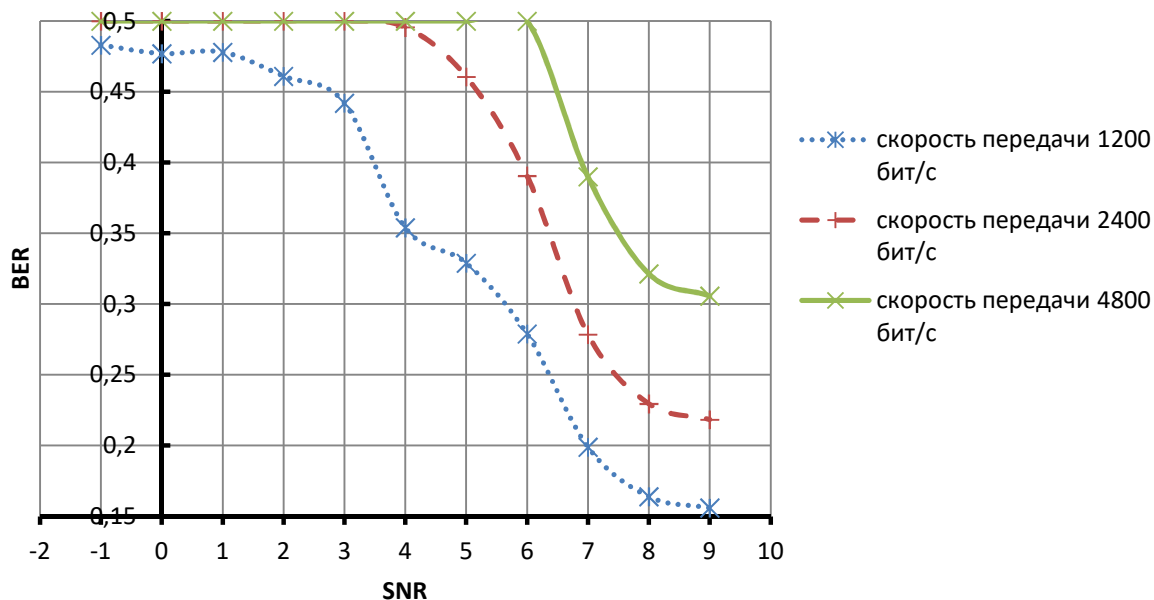


Рисунок 24 - Зависимость BER от SNR для сигнала уровнем 0,8В

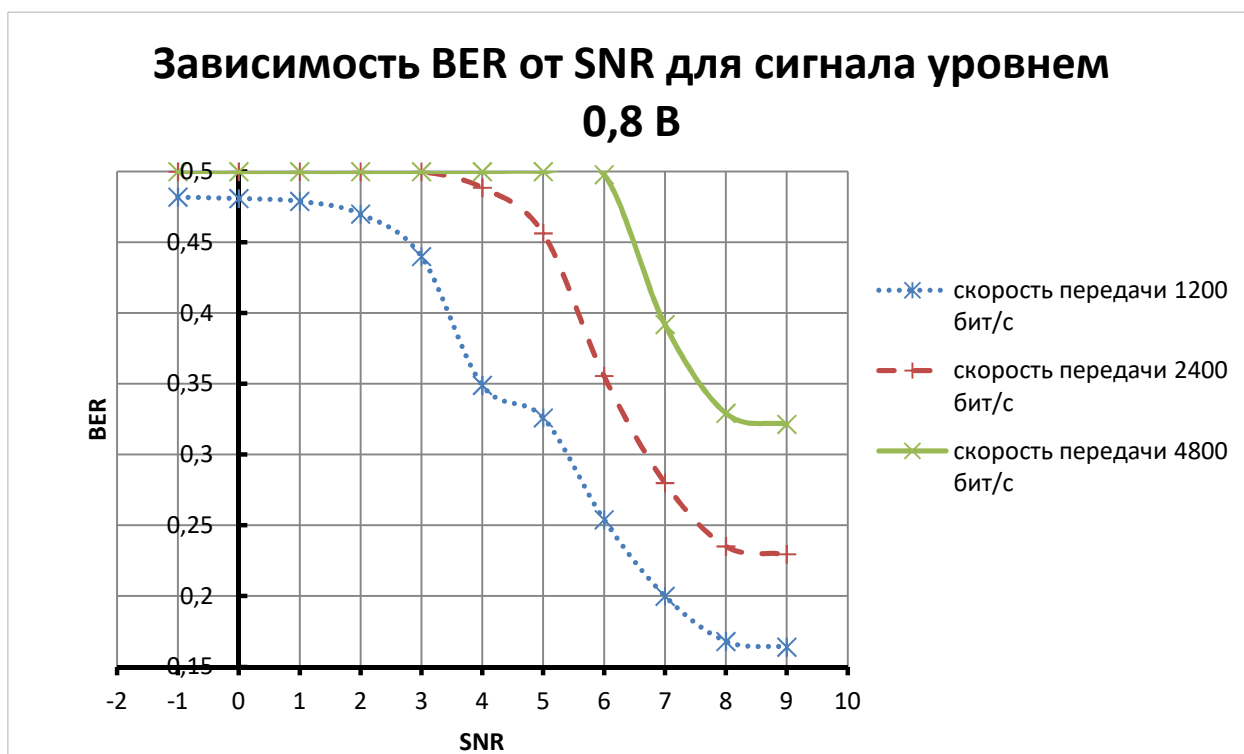


Рисунок 25 - Зависимость BER от SNR для сигнала уровнем 2,2 В

Результаты исследования помехоустойчивости АПД с использованием узлов обеспечения целостности к влиянию аддитивного гармонической помехи приведены в таблице 5.

Таблица 5 - Результаты исследования помехоустойчивости АПД с использованием узлов обеспечения целостности

Скорость передачи	BER								
	1200			2400			4800		
Уровень сигнала	0,36	0,8	2,2	0,36	0,8	2,2	0,36	0,8	2,2
SNR, дБ									
-1	0,459	0,455	0,448	0,5	0,5	0,5	0,5	0,5	0,5
0	0,442	0,435	0,442	0,5	0,5	0,5	0,5	0,5	0,5
1	0,253	0,26	0,234	0,354	0,364	0,328	0,496	0,5	0,459
2	0,153	0,165	0,162	0,214	0,231	0,227	0,3	0,323	0,318
3	0,09	0,113	0,122	0,13	0,159	0,171	0,182	0,222	0,239

Продолжение таблицы 5

	BER								
Скорость передачи	1200			2400			4800		
Уровень сигнала	0,36	0,8	2,2	0,36	0,8	2,2	0,36	0,8	2,2
SNR, дБ									
4	0,053	0,065	0,058	0,074	0,091	0,081	0,104	0,127	0,114
5	0,033	0,03	0,041	0,05	0,04	0,058	0,065	0,058	0,08
6	0,013	0,019	0,015	0,018	0,027	0,021	0,026	0,037	0,03
7	0,007	0,014	0,005	0,01	0,02	0,002	0,013	0,028	0,002
8	0,003	0,005	0,003	0,004	0,007	0,007	0,006	0,01	0,01
9	0,000	0,003	0,002	0,001	0,001	0,0001	0,0006	0,002	0,0001

По данным таблицы 5 построены графики зависимости коэффициента ошибок на бит BER от соотношения сигнал/шум SNR, приведенные на рисунках 26 - 31.

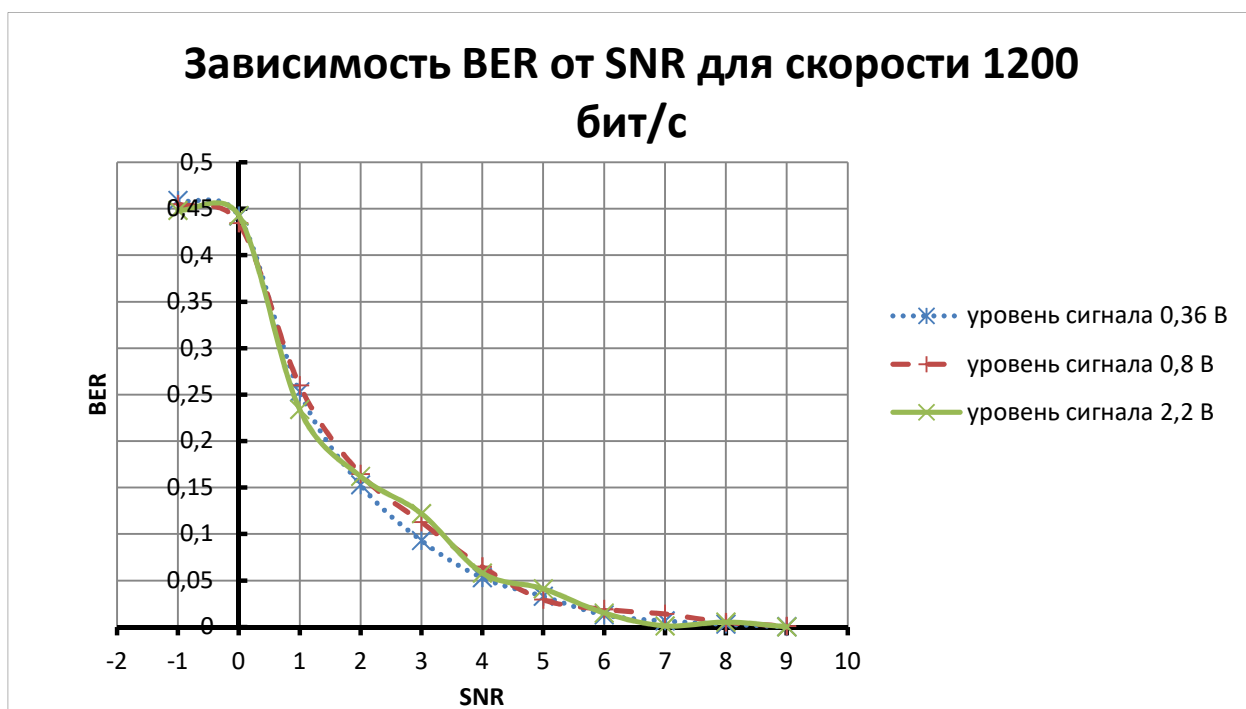


Рисунок 26- Зависимость BER от SNR для скорости 1200 бит/с

Зависимость BER от SNR для скорости 2400 бит/с

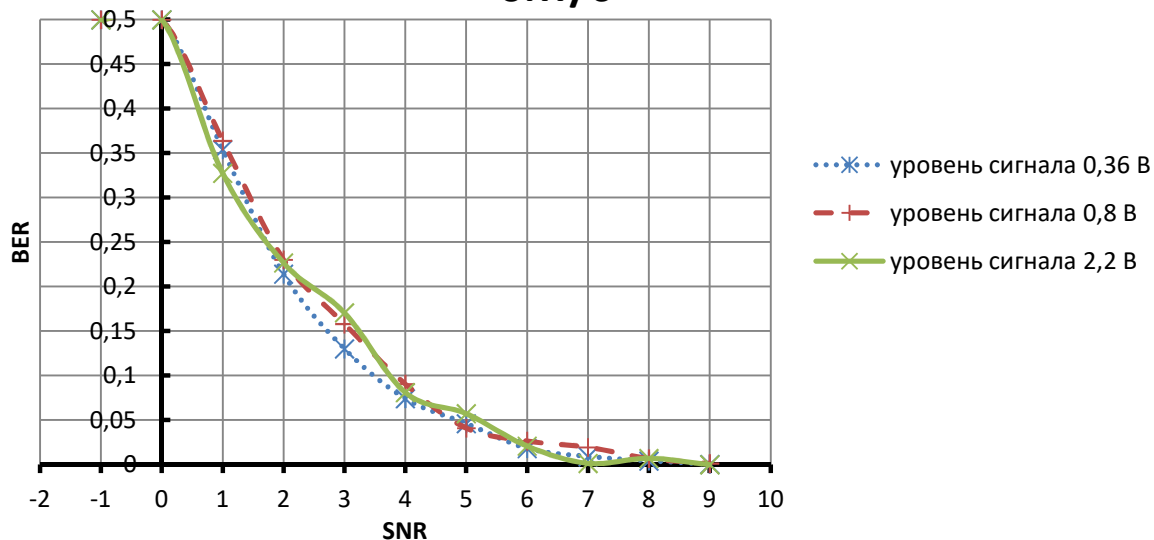


Рисунок 27- Зависимость BER от SNR для скорости 2400 бит/с

Зависимость BER от SNR для скорости 4800 бит/с

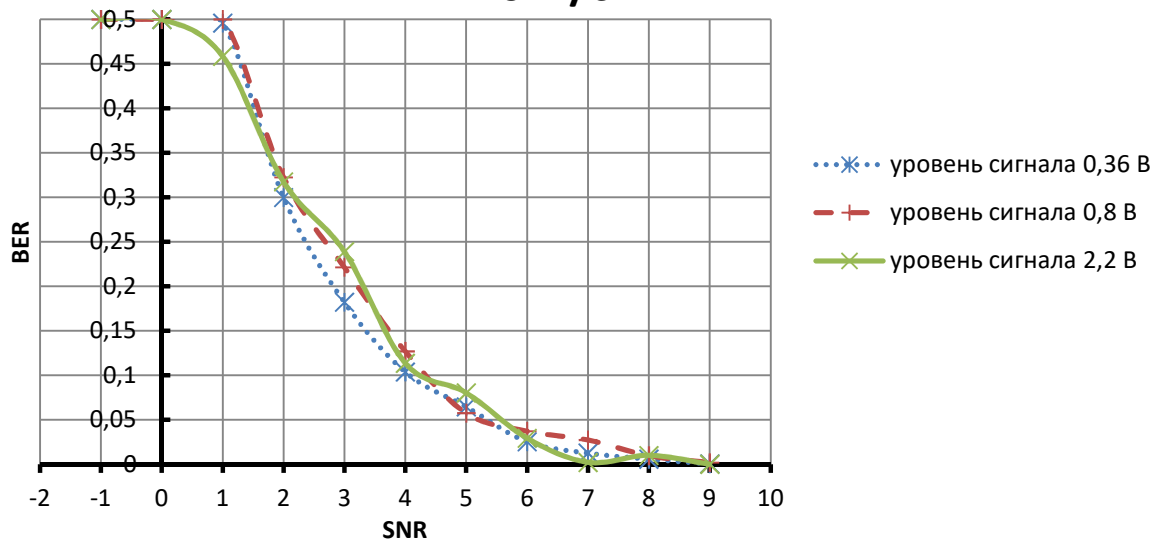


Рисунок 28 - Зависимость BER от SNR для скорости 4800 бит/с

Зависимость BER от SNR для сигнала уровнем 0,36 В

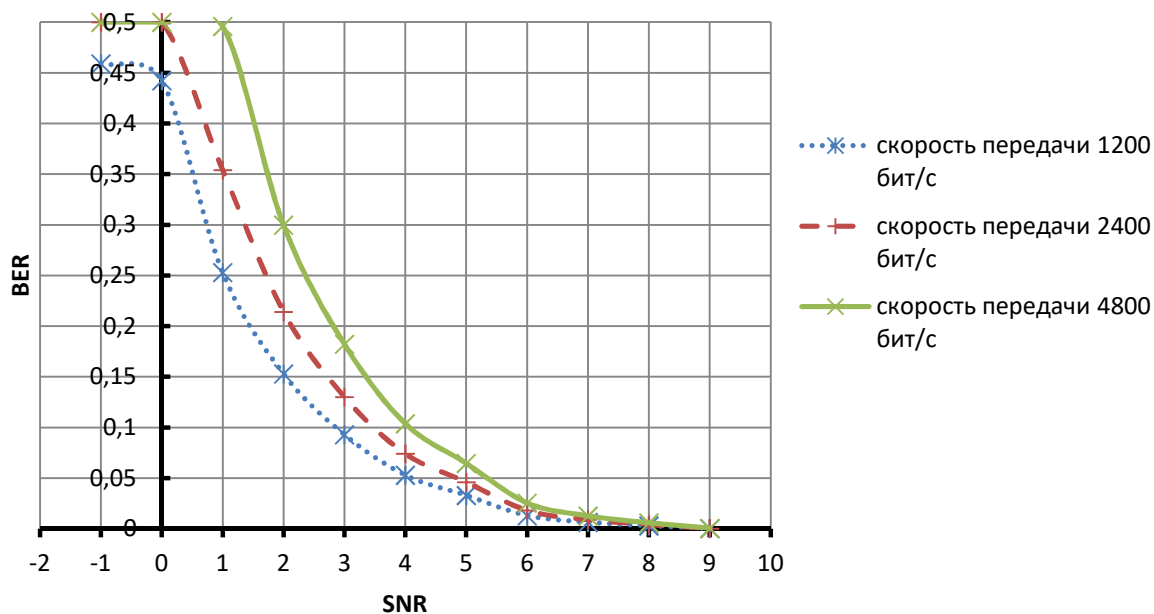


Рисунок 29 - Зависимость BER от SNR для сигнала уровнем 0,36В

Зависимость BER от SNR для сигнала уровнем 0,8 В

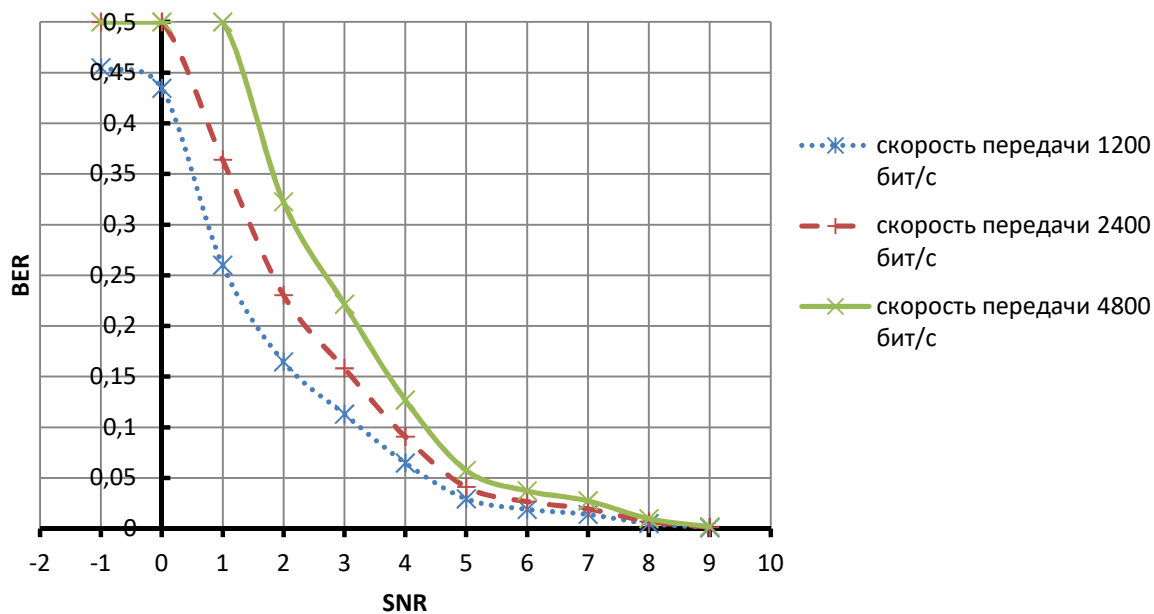


Рисунок 30 - Зависимость BER от SNR для сигнала уровнем 0,8В

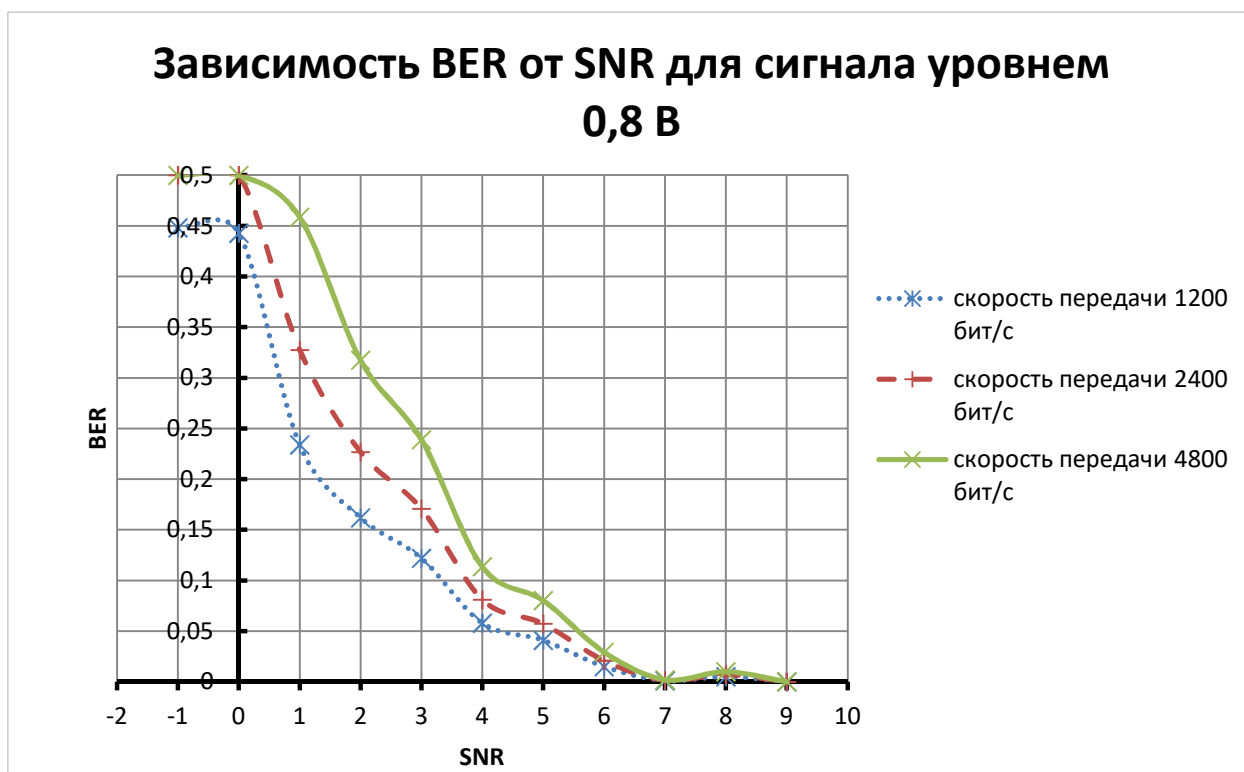


Рисунок 31 - Зависимость BER от SNR для сигнала уровнем 2,2В

На рисунках 26-28 приведена зависимость BER от SNR при постоянной скорости передачи. Как видно из рисунков, изменение уровня сигнала незначительно влияет на значение коэффициента BER. Это связано с тем, что при проведении исследований задавалось значение отношения сигнал/шум, а не значение шума. Для разного уровня сигнала, при одном и том же значении коэффициента SNR, уровень шума разный.

На рисунках 29-31 приведена зависимость BER от SNR при постоянном уровне сигнала. Следует обратить внимание на наличие связи между скоростью передачи информации и помехоустойчивостью. Увеличение скорости передачи данных (при фиксированной полосе частот канала и мощности передатчика) приводит к уменьшению помехоустойчивости. И наоборот, снижение скорости передачи информации (при фиксированной полосе частот канала и мощности передатчика) приводит к увеличению помехоустойчивости.

Исследование показало, что узлы обеспечения целостности АПД одинаково функционируют при разном уровне сигнала и скорости передачи данных.

3.2.2 Исследование помехоустойчивости АПД при заданном уровне шума

Результаты исследования помехоустойчивости АПД без узлов обеспечения целостности к влиянию аддитивного белого гауссова шума приведены в таблице 6.

Таблица 6 - Результаты исследования помехоустойчивости АПД без узлов обеспечения целостности

	BER								
Скорость передачи	1200			2400			4800		
Уровень сигнала	0,36	0,8	2,2	0,36	0,8	2,2	0,36	0,8	2,2
Ршума									
0,45	0,482	0,442	0,2	0,5	0,5	0,28	0,5	0,5	0,392
0,36	0,48	0,354	0,168	0,5	0,496	0,235	0,5	0,5	0,329
0,29	0,479	0,329	0,164	0,5	0,461	0,23	0,5	0,5	0,322
0,23	0,475	0,279	0,157	0,5	0,391	0,199	0,5	0,5	0,319
0,18	0,445	0,199	0,152	0,5	0,279	0,164	0,5	0,39	0,269
0,14	0,359	0,164	0,145	0,5	0,269	0,156	0,5	0,322	0,212
0,11	0,319	0,156	0,138	0,447	0,212	0,147	0,5	0,306	0,174
0,09	0,269	0,147	0,131	0,377	0,174	0,141	0,5	0,275	0,155
0,07	0,212	0,141	0,124	0,297	0,155	0,134	0,416	0,247	0,147
0,06	0,174	0,134	0,116	0,244	0,11	0,127	0,341	0,21	0,141
0,05	0,155	0,127	0,1	0,217	0,07	0,112	0,304	0,17	0,134

По данным, приведенным в таблице 6 построенным графики зависимости коэффициента ошибок на бит BER от соотношения сигнал/шум $P_{\text{шума}}$, приведенные на рисунках 32 – 34

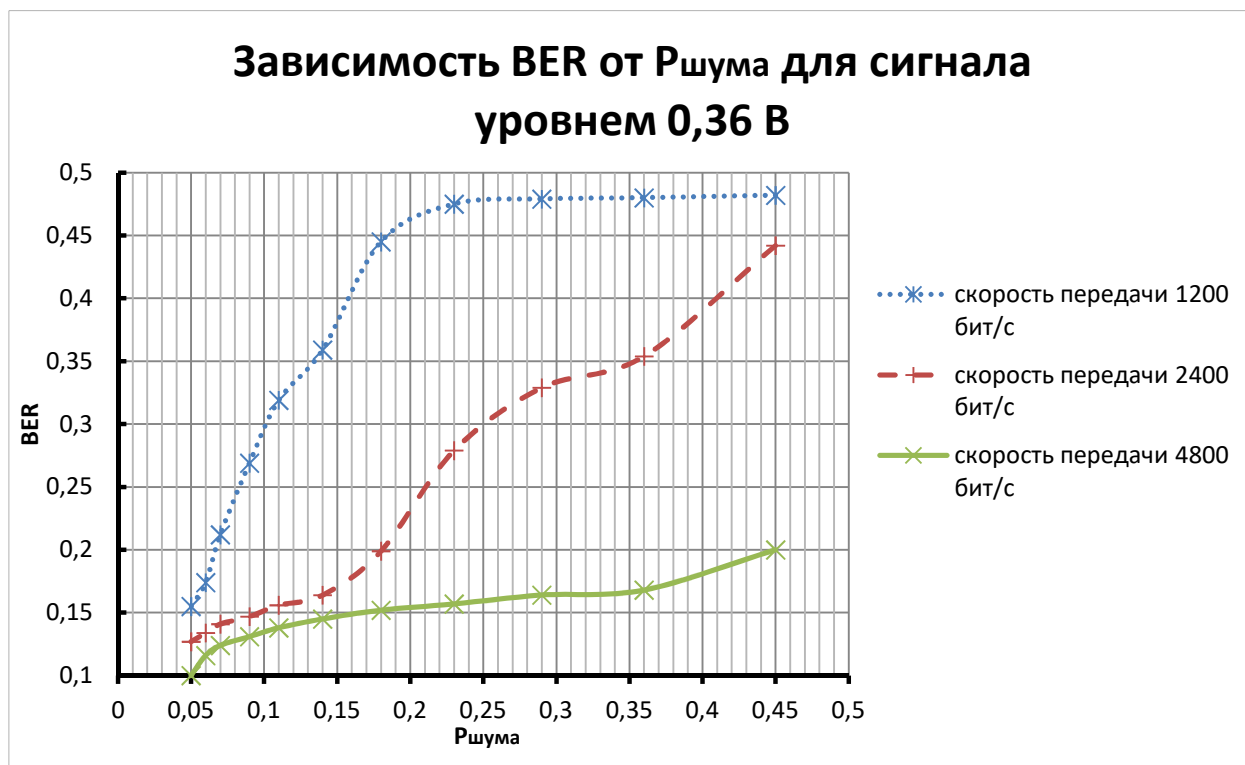


Рисунок 32 - Зависимость BER от $P_{\text{шума}}$ для сигнала уровнем 0,36В

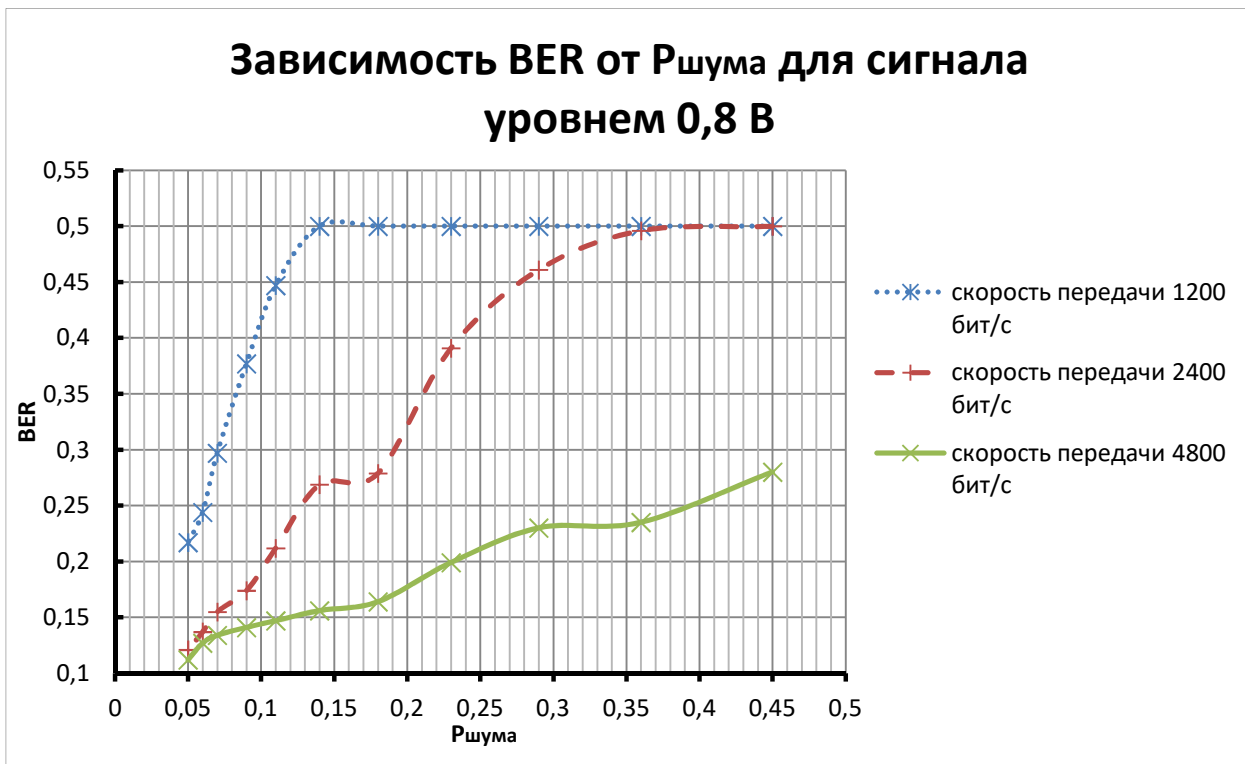


Рисунок 33 - Зависимость BER от $P_{\text{шума}}$ для сигнала уровнем 0,8В

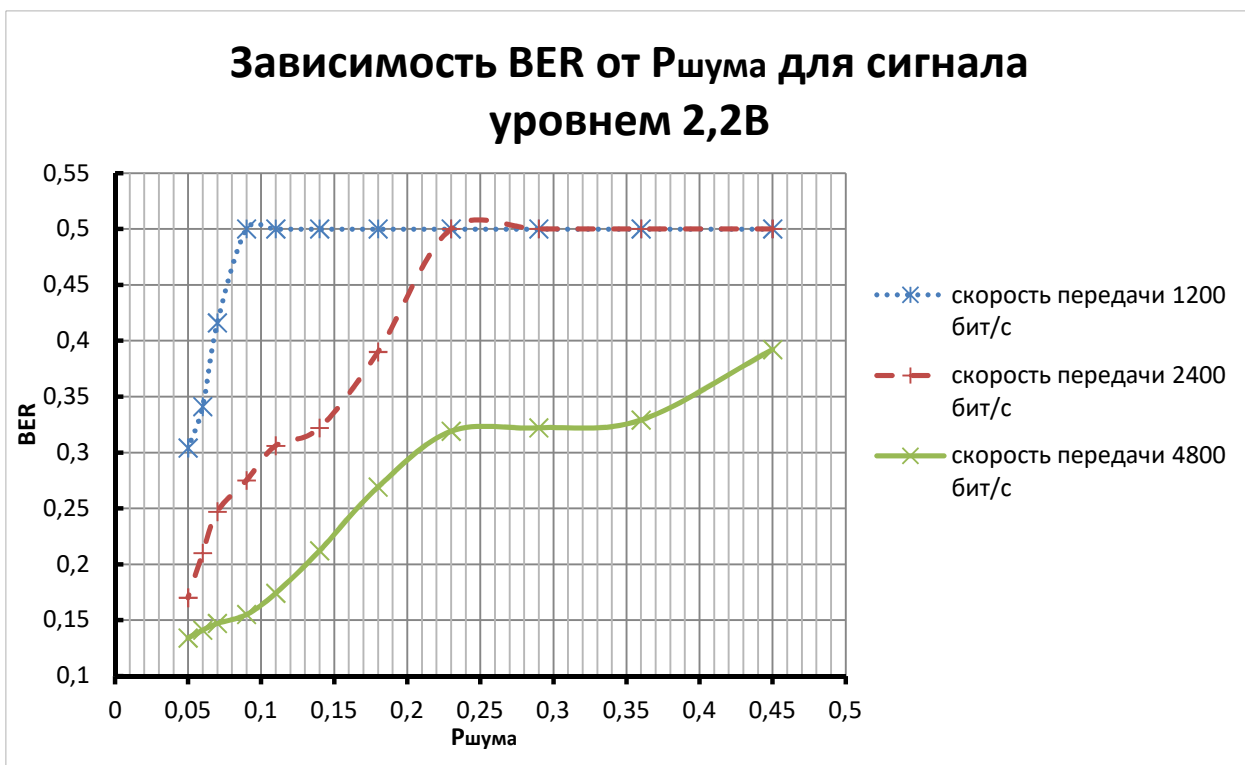


Рисунок 34 - Зависимость BER от $P_{\text{шума}}$ для сигнала уровнем 2,2В

На рисунках 32-34 приведена зависимость BER от $P_{\text{шума}}$ при постоянной скорости передачи. Как видно из рисунков, изменение уровня сигнала влияет на значение коэффициента BER. Чем выше уровень сигнала, тем меньше коэффициент BER, так как при одинаковом уровне шума и разном уровне сигнала, отношение сигнал /шум разное, что подтверждается значениями из таблицы 7.

Таблица 7 – Отношение сигнал/шум

Ршума	0,45	0,36	0,29	0,23	0,18	0,14	0,11	0,09	0,07	0,06	0,05
Уровень сигнала 0,36В	-1	0	1	2	3	4	5	6	7	8	9
Уровень сигнала 0,8В	2,5	3,5	4,5	5,5	6,5	7,5	8,5	9,5	10,5	11,5	12,5
Уровень сигнала 2,2В	6,9	7,9	8,9	9,9	10,9	11,9	12,9	13,9	14,9	15,9	16,9

Значение отношения сигнал/шум, рассчитанное по формуле 9, находится на пересечении значения уровня шума ($P_{\text{шума}}$) и уровня сигнала. Для большей наглядности таблица 7 иллюстрируется рисунком 35, из которого видно, что при уровне сигнала 0,36В, значение коэффициента SNR практически в 2 раза меньше этого же значения при уровне сигнала 2,2В.

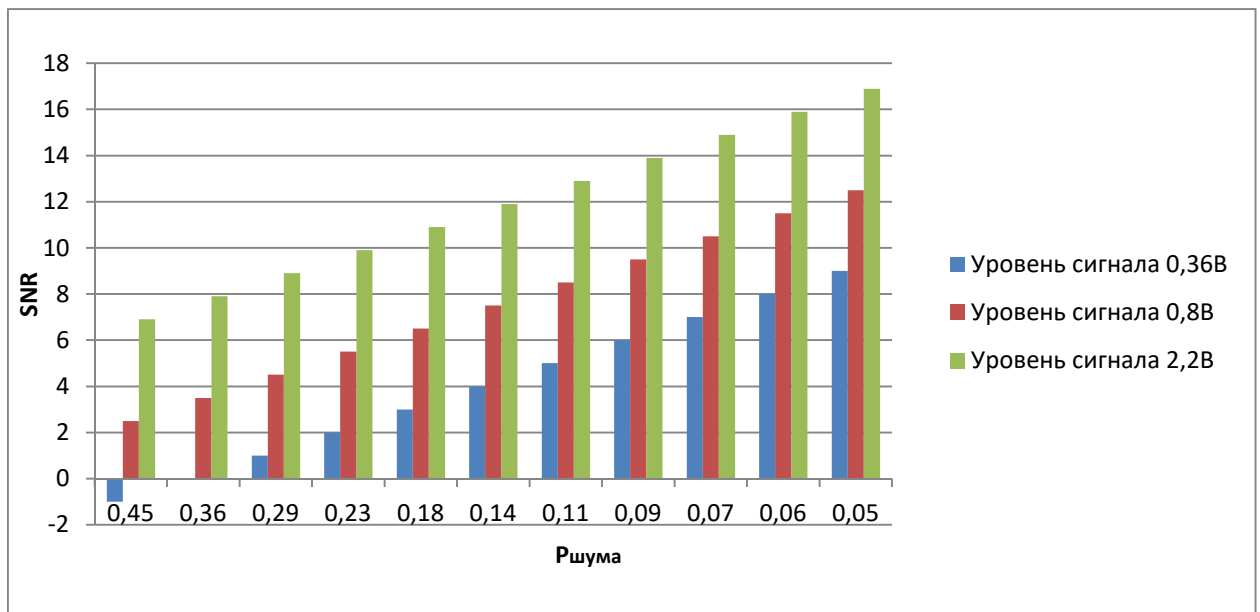


Рисунок 35 - Отношение сигнал/шум

На рисунках 36-38 приведена зависимость BER от $P_{\text{шума}}$ при постоянном уровне сигнала. Следует обратить внимание на наличие связи между скоростью передачи данных, шумом и уровнем ошибок. Наличие шума приводит к изменению одного или нескольких битов. При увеличении скорости передачи данных биты передаются быстрее поэтому при одинаковом уровне шума и большей скорости передачи данных появляется большее количество «поврежденных» битов, что подтверждается теоремой Шеннона.

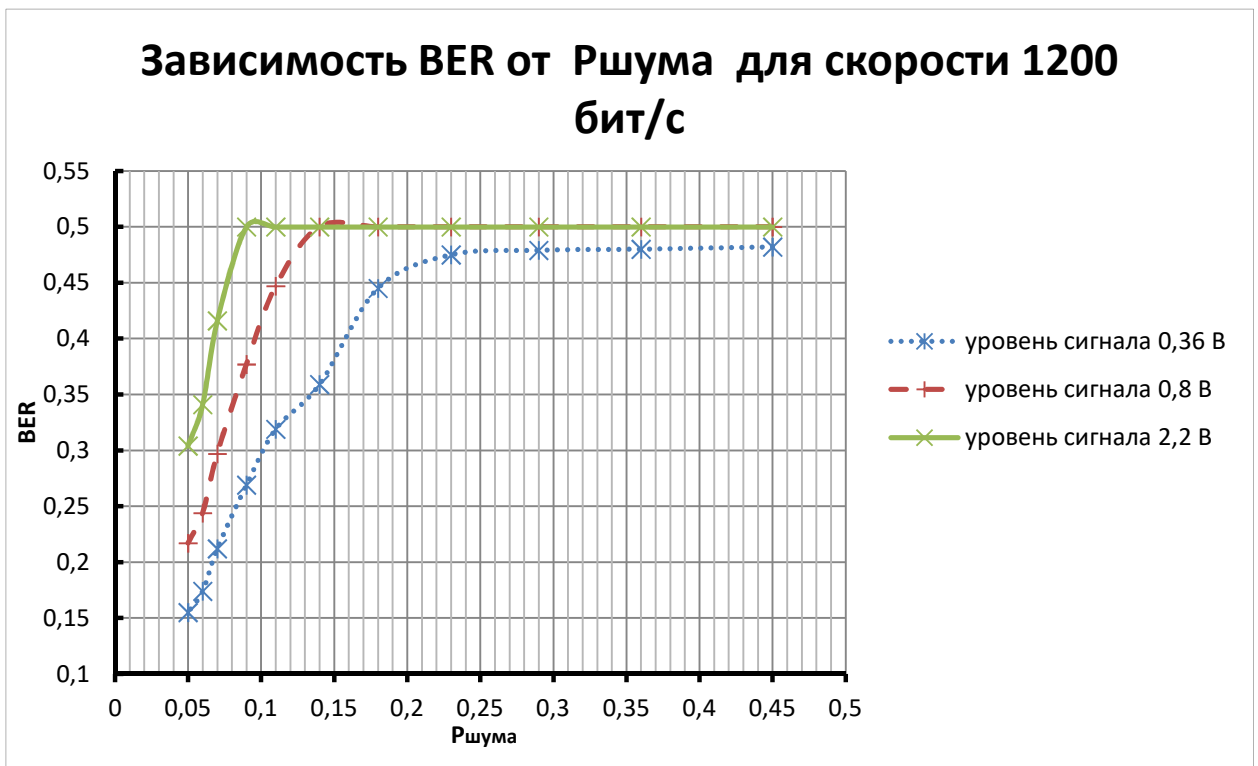


Рисунок 36 - Зависимость BER от $P_{\text{шума}}$ для скорости 1200 бит/с

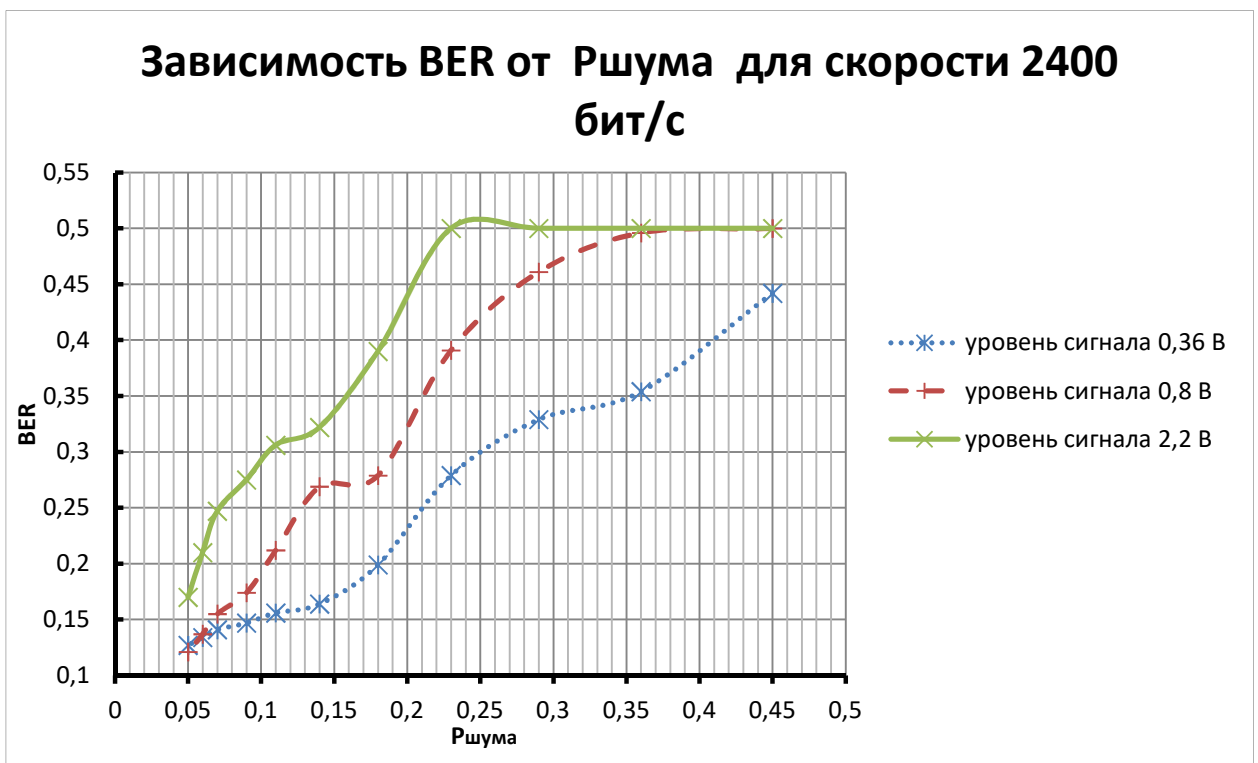


Рисунок 37 - Зависимость BER от $P_{\text{шума}}$ для скорости 2400 бит/с

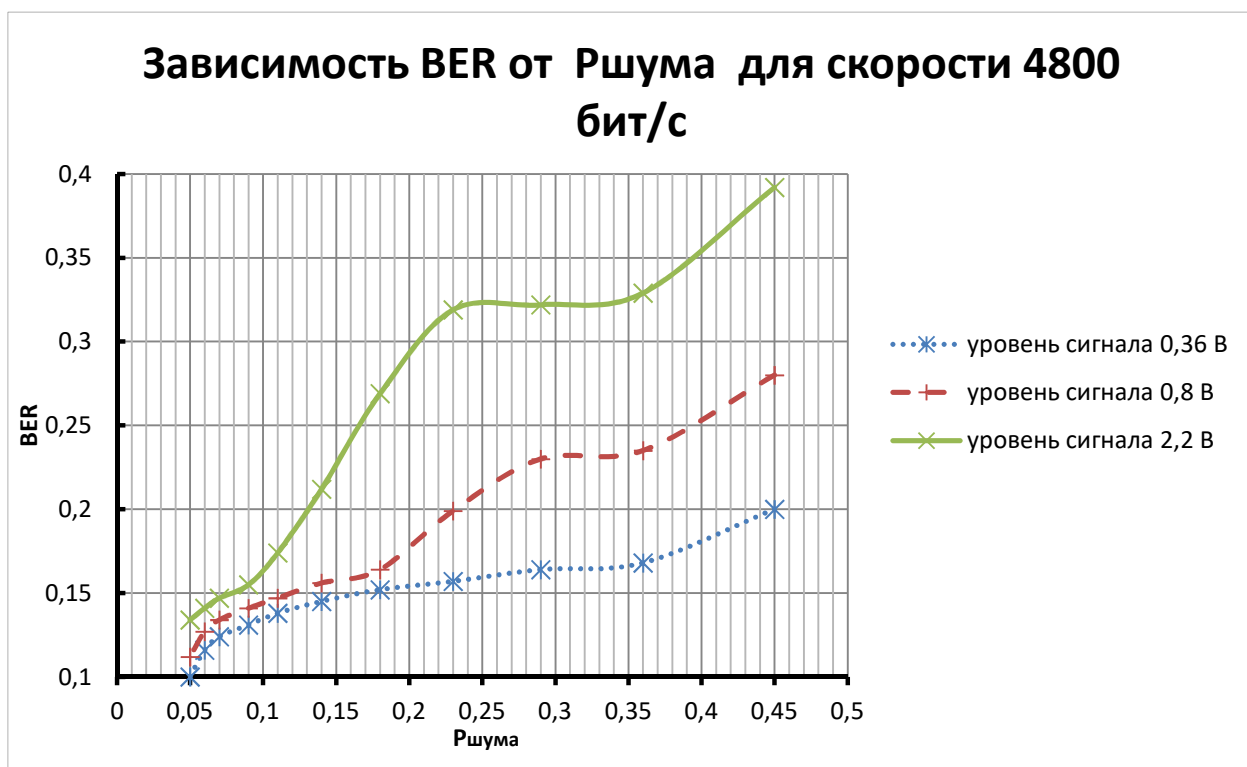


Рисунок 38 - Зависимость BER от Ршума для скорости 4800 бит/с

Результаты исследования помехоустойчивости АПД с использованием узлов обеспечения целостности к влиянию аддитивного гармонической помехи приведены в таблице 8.

Таблица 8 - Результаты исследования помехоустойчивости АПД с использованием узлов обеспечения целостности

	BER								
	1200			2400			4800		
Скорость передачи									
Уровень сигнала	0,36	0,8	2,2	0,36	0,8	2,2	0,36	0,8	2,2
Уровень шума									
0,45	0,459	0,113	0,015	0,5	0,1582	0,021	0,5	0,2215	0,0294
0,36	0,442	0,065	0,0051	0,5	0,0907	0,0071	0,5	0,127	0,01
0,29	0,253	0,03	0,0031	0,354	0,0412	0,0043	0,496	0,0576	0,0061

Продолжение таблицы 8

Скорость передачи	1200			2400			4800		
Уровень сигнала	0,36	0,8	2,2	0,36	0,8	2,2	0,36	0,8	2,2
Уровень шума									
0,23	0,153	0,019	0,0021	0,214	0,0266	0,0029	0,3	0,0372	0,0041
0,18	0,09	0,014	0,0017	0,13	0,0196	0,0021	0,182	0,0274	0,003
0,14	0,053	0,005	0,0012	0,074	0,007	0,0016	0,104	0,0098	0,0024
0,11	0,033	0,003	0,00109	0,05	0,0042	0,0014	0,065	0,0059	0,0019
0,09	0,013	0,0024	0,00072	0,018	0,0034	0,001	0,026	0,0051	0,0011
0,07	0,007	0,002	0,00053	0,01	0,0026	0,00071	0,013	0,00451	0,0007
0,06	0,003	0,0015	0,00024	0,004	0,0015	0,00047	0,006	0,0032	0,0003
0,05	0	0,001	0,00018	0,001	0,001	0,0003	0,0006	0,0027	0,00024

По данным таблицы 8 построены графики зависимости коэффициента ошибок на бит BER от $P_{\text{шума}}$, приведенные на рисунках 39 - 47.

На рисунках 39-41 приведена зависимость BER от $P_{\text{шума}}$ при постоянной скорости передачи. Как видно из рисунков, изменение уровня сигнала влияет на значение коэффициента BER. Чем выше уровень сигнала, тем меньше коэффициент BER, так как при одинаковом уровне шума и разном уровне сигнала, отношение сигнал /шум разное.

Зависимость BER от $P_{\text{шума}}$ для сигнала уровнем 0,36 В

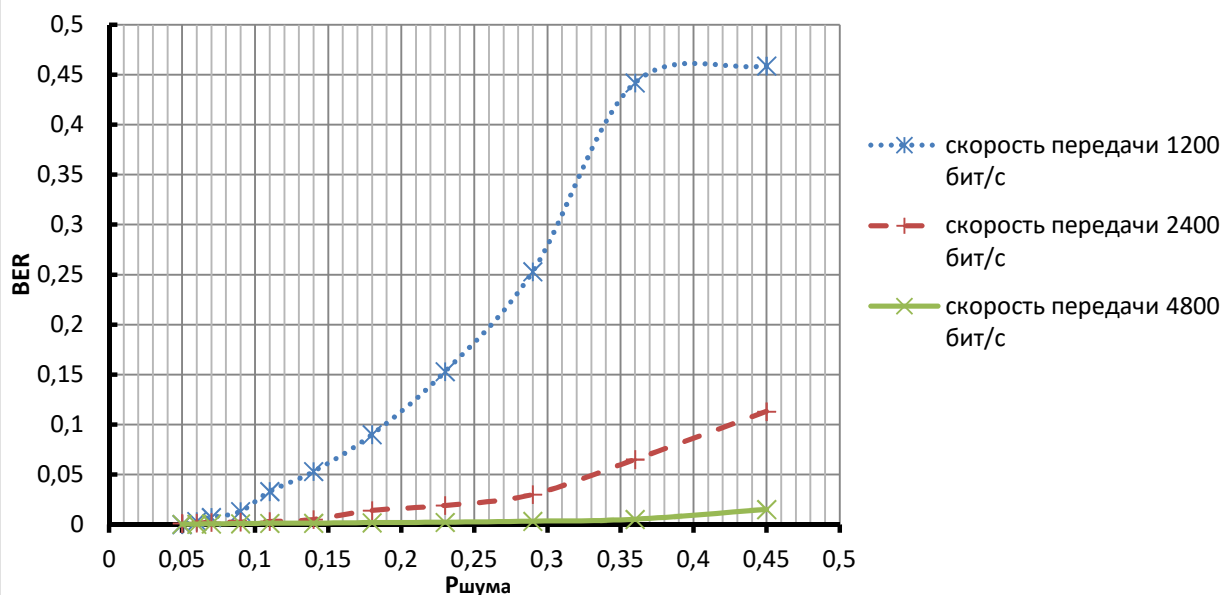


Рисунок 39 - Зависимость BER от $P_{\text{шума}}$ для сигнала уровнем 0,36В

Зависимость BER от $P_{\text{шума}}$ для сигнала уровнем 0,8 В

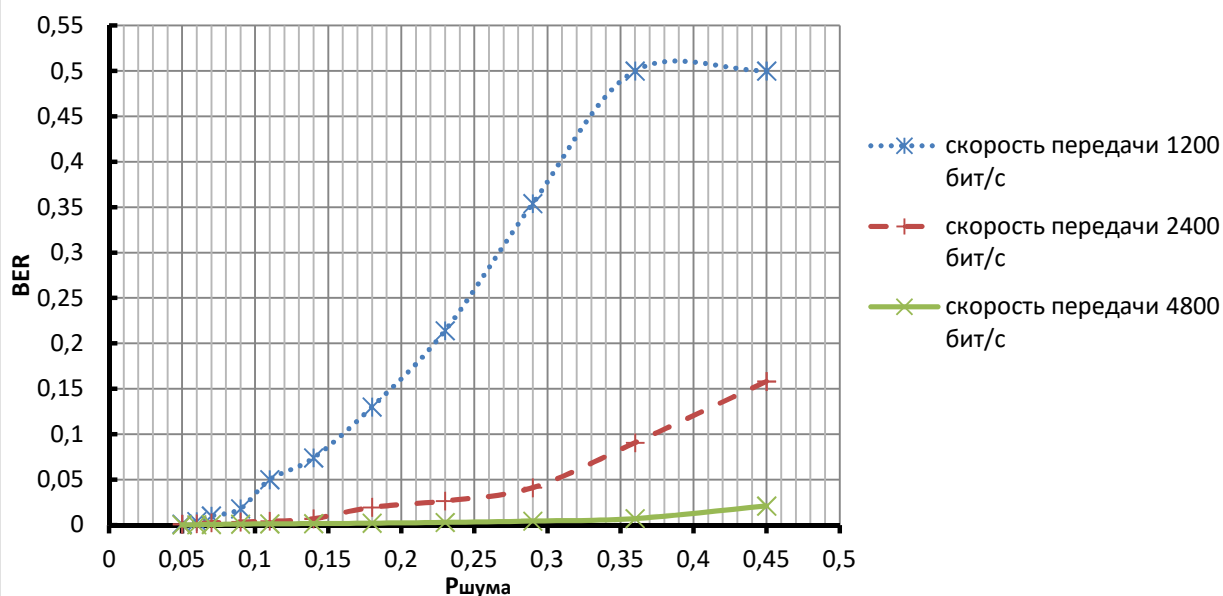


Рисунок 40 - Зависимость BER от $P_{\text{шума}}$ для сигнала уровнем 0,8В

Зависимость BER от $P_{\text{шума}}$ для сигнала уровнем 2,2В

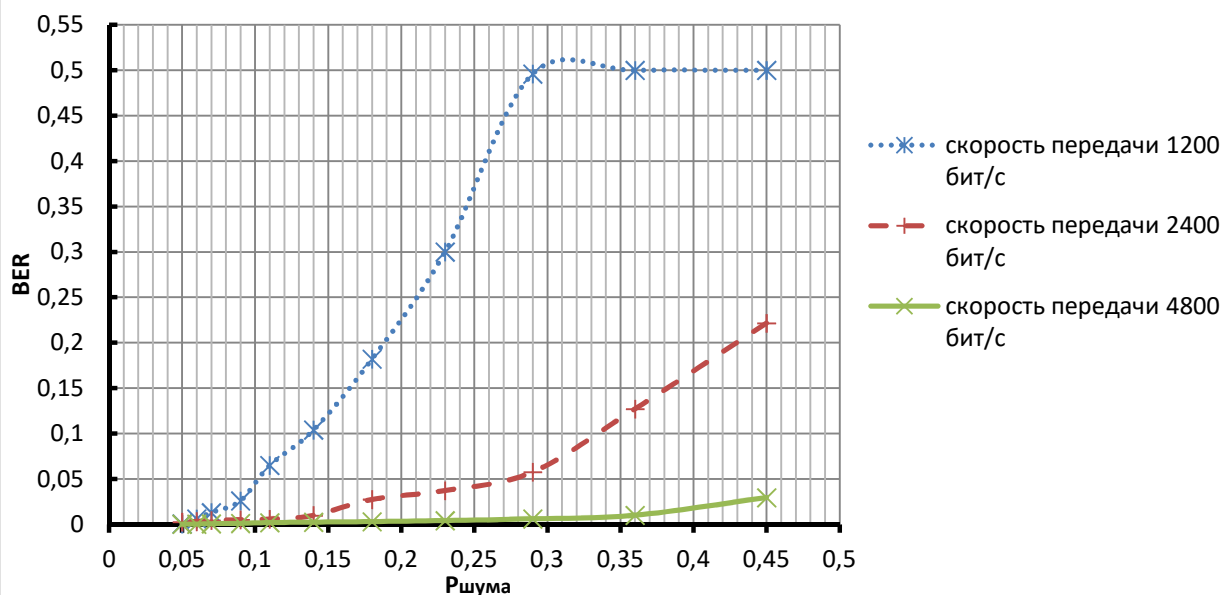


Рисунок 41 - Зависимость BER от $P_{\text{шума}}$ для сигнала уровнем 2,2В

Зависимость BER от $P_{\text{шума}}$ для скорости 1200 бит/с

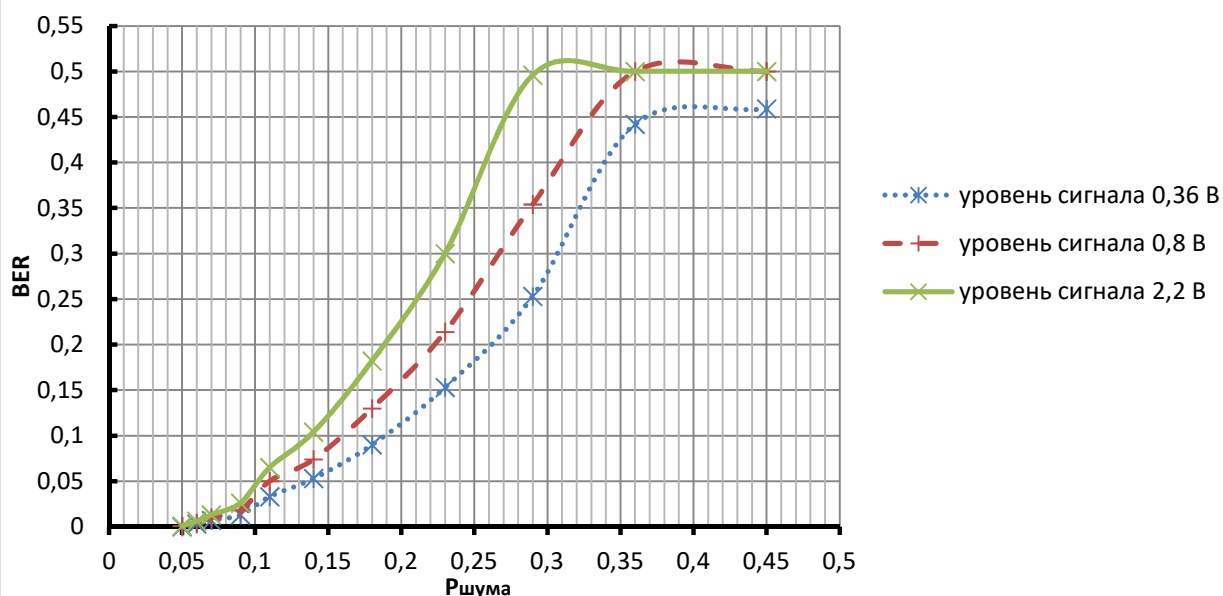


Рисунок 42 - Зависимость BER от $P_{\text{шума}}$ для скорости 1200 бит/с

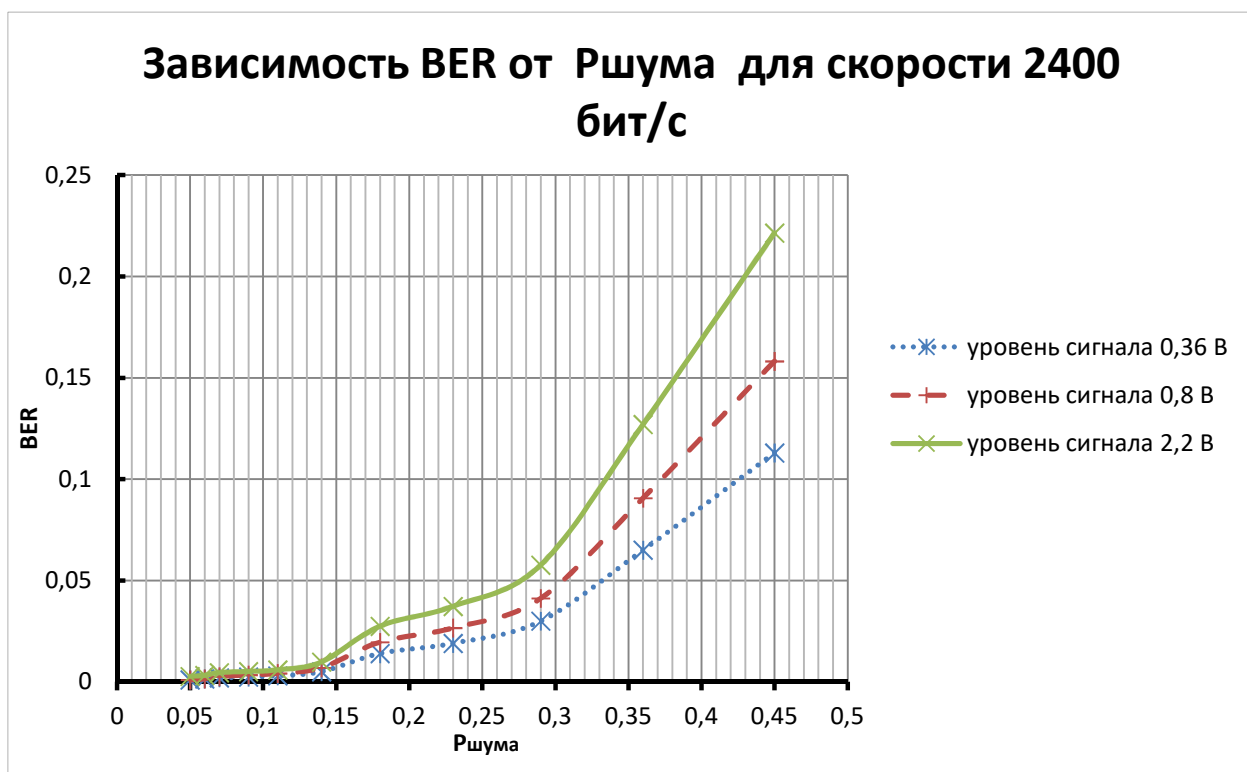


Рисунок 43 - Зависимость BER от Ршума для скорости 2400 бит/с

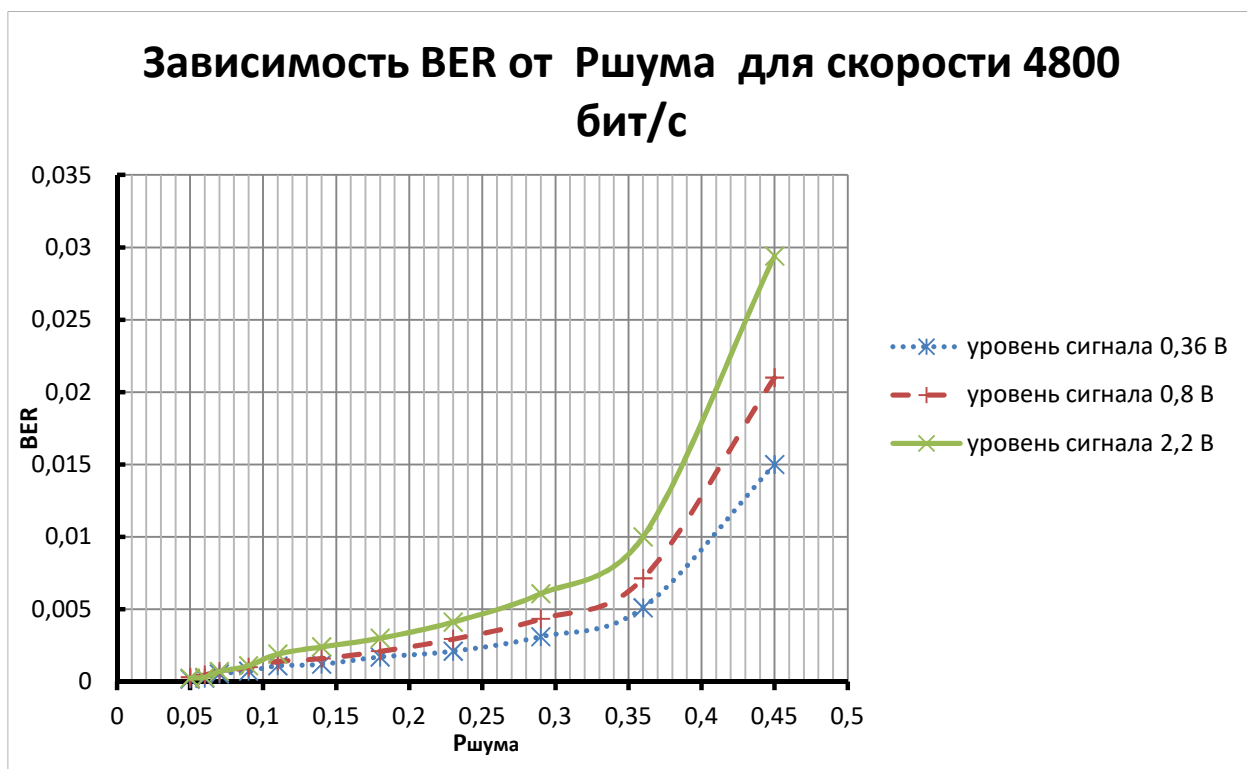


Рисунок 44 - Зависимость BER от Ршума для скорости 4800 бит/с

На рисунках 42-44 приведена зависимость BER от $P_{\text{шума}}$ при постоянной скорости передачи. Из рисунков видно наличие связи между скоростью передачи данных, шумом и уровнем ошибок. Наличие шума приводит к изменению одного или нескольких битов. При увеличении скорости передачи данных биты передаются быстрее поэтому при одинаковом уровне шума и большей скорости передачи данных появляется большее количество «поврежденных» битов, что подтверждается теоремой Шеннона.

Исследование показало, что использование узлов обеспечения целостности позволило уменьшить влияние гармонической помехи на информативный сигнал. Данный вывод проиллюстрировано рисунками 32-34.

Так как изменение соотношения сигнал/шум незначительно влияет на значение коэффициента BER, то на приведенных диаграммах показан один уровень SIN и разная скорость передачи:

- рисунок 45 - уровень соотношения сигнал/шум составляет 0,36 В, скорость передачи 1200 бит/с;
- рисунок 46 - уровень соотношения сигнал/шум составляет 0,36 В, скорость передачи 2400 бит/с;
- рисунок 47 - уровень соотношения сигнал/шум составляет 0,36 В, скорость передачи 4800 бит/с.

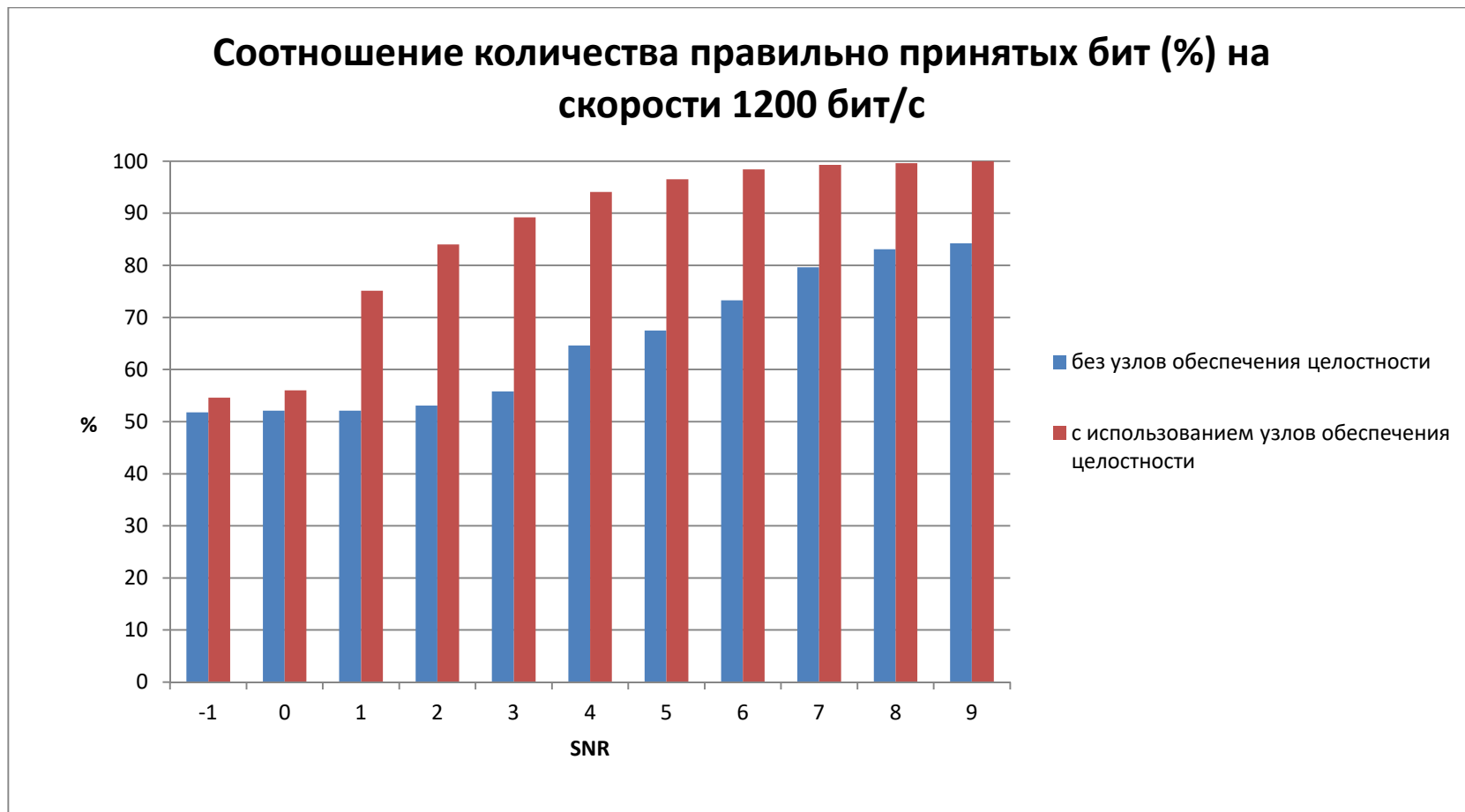


Рисунок 45 – Соотношение количества правильно принятых бит (%) на скорость 1200 бит/с

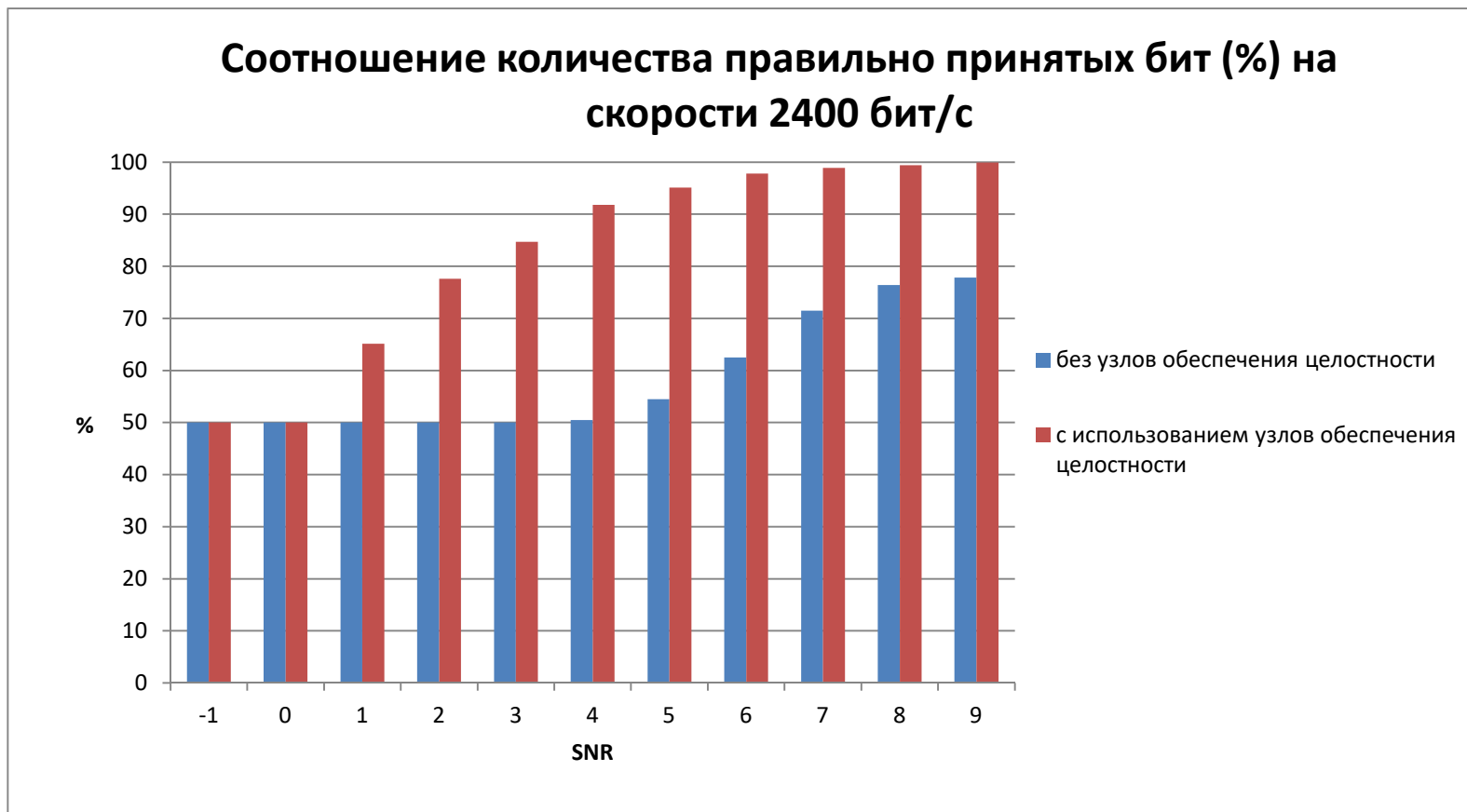


Рисунок 46 – Соотношение количества правильно принятых бит (%) на скорость 2400 бит/с

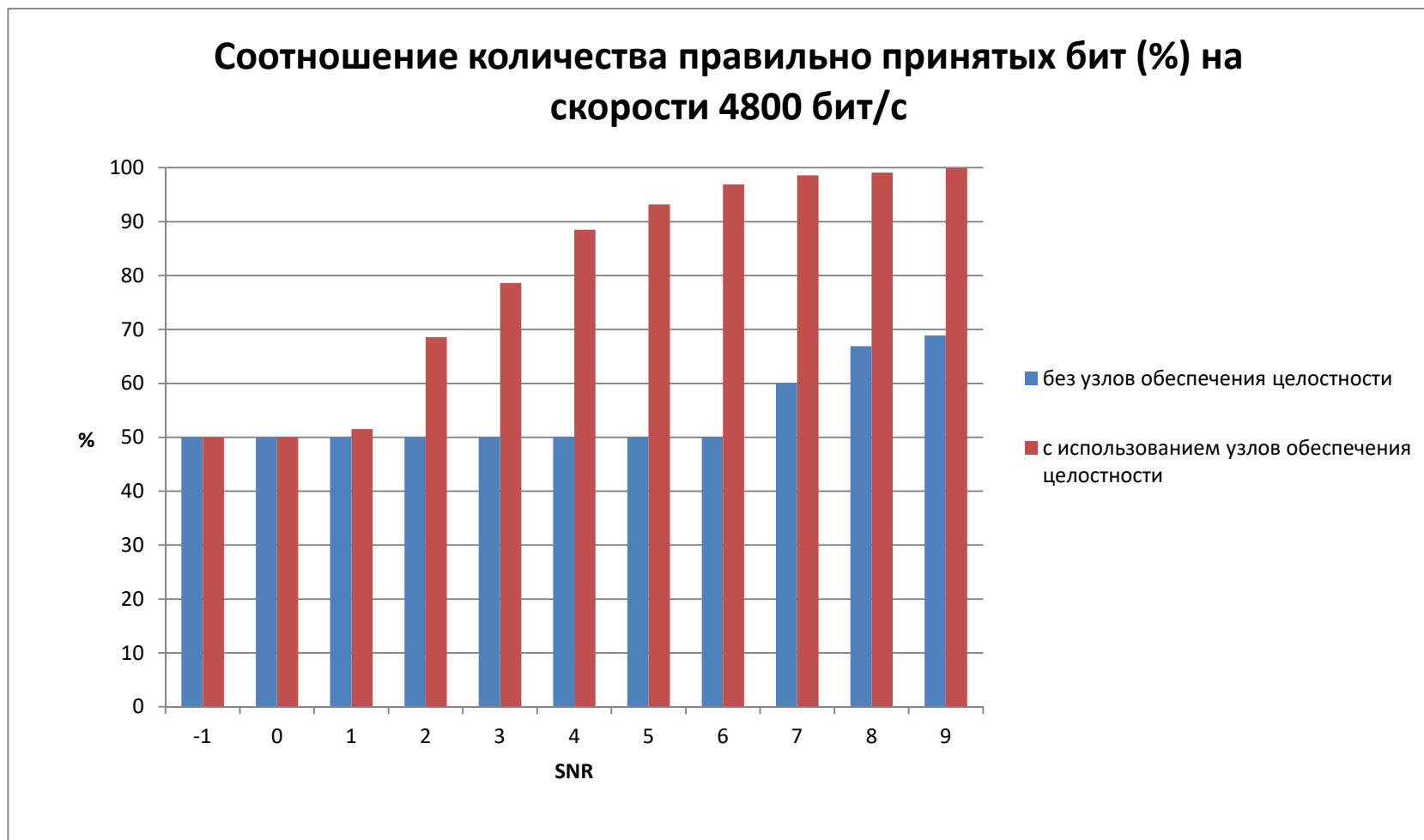


Рисунок 47 – Соотношение количества правильно принятых бит (%) на скорость 4800 бит/с

ЗАКЛЮЧЕНИЕ

В ходе выполнения дипломного проекта были решены следующие задачи:

- проведен анализ факторов, влияющих на нарушение целостности данных;
- разработаны узлы целостности данных для факторов, влияющих на нарушение целостности, выявленных в процессе анализа;
- проведены исследования помехоустойчивости аппаратуры передачи данных без использования узлов обеспечения целостности данных;
- проведены исследования помехоустойчивости аппаратуры передачи данных с использованием узлов обеспечения целостности данных.

В результате дипломного проектирования все пункты технического задания были выполнены, были разработаны обеспечения целостности.

Результат выполнения работы используется в рабочем процессе АО НПП «Рубин», что подтверждено актом о внедрении.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Доктрина информационной безопасности Российской Федерации от 5.12.2016 [Электронный ресурс]. – Режим доступа свободный: <http://www.scrf.gov.ru/documents/6/5.html>;
- 2 Военное обозрение. Вооружение. Системы ПВО. ПВО России приобретает клыки – комплект средств автоматизации «Барнаул-Т» [Электронный ресурс]. – Режим доступа свободный: <http://topwar.ru/7504-pvo-rossii-priobretaet-klyki-komplekt-sredstv-avtomatizacii-barnaul-t.html>;
- 3 ГОСТ 15.971-90 «Системы обработки информации. Термины и определения»;
- 4 ФЗ РФ от 27.07.2006 №149 - ФЗ «Об информации, информационных технологиях и о защите информации»;
- 5 Брескина О.М., Корешкова А.А., Иванов А.П., Реализация генератора псевдослучайной последовательности на ПЛИС фирмы Altera // Многоядерные процессоры, параллельное программирование, ПЛИС, системы обработки сигналов. – 2016. – С.17 – 20;
- 6 ГОСТ Р 50754-95 «Язык описания аппаратуры цифровых систем –VHDL. Описание языка»;
- 7 АЕЯР.431260.734ТУ «Микросхемы интегральные 5576ХС4Т»;
- 8 Р 50.1.056-2005 «Техническая защита информации. Основные термины и определения»;
- 9 Харкевич А.А. Борьба с помехами 2изд., М., 1960;
- 10 ГОСТ 17422-82 «Системы передачи данных. Скорость передачи данных и основные параметры помехоустойчивых циклических кодов»;
- 11 ГОСТ 28082-89 «Системы обработки информации. Методы обнаружения ошибок при последовательной передаче данных»;
- 12 Помехи, типы помех, методы борьбы с помехами [Электронный ресурс]. – Режим доступа свободный: http://koralexand.ru/?page_id=105;

13 Журнал сетевых решений LAN [Электронный ресурс]. – Режим доступа свободный: [http:// www.osp.ru/lan/2005/377334](http://www.osp.ru/lan/2005/377334)

ПРИЛОЖЕНИЕ А

(обязательное)

Код программы, реализующей защиту от дребезга

```
ENTITY ANTIDR IS
    PORT
    (
        CLK                : IN STD_LOGIC;
        IN_SIGNAL          : IN STD_LOGIC;
        OUT_SIGNAL         : OUT STD_LOGIC
    );
END;
ARCHITECTURE be OF ANTIDR IS
    signal      sig : std_logic;
    signal cnt: integer range 0 to 20000;
BEGIN
    process (CLK)
    begin
        If IN_SIGNAL = '1' then
            If CLK = '1' and CLK'event then
                if cnt /= 18342 then
                    cnt <= cnt+1;
                elsif cnt = 18342 then
                    if IN_SIGNAL = '1' then
                        OUT_SIGNAL <= IN_SIGNAL;
                    end if;
                end if;
            end if;
        else
            cnt <=0;
            OUT_SIGNAL <= '0';
        end if;
    end process;
END;
```

ПРИЛОЖЕНИЕ Б

(обязательное)

Код программы, реализующей имитатор стыка RS-232

```
library ieee;
use ieee.Std_Logic_1164.all;
USE IEEE.numeric_std.ALL;
entity UART_1 is
    generic (UART_Speed : integer := 38400 ;    -- UART Speed (;default = 9600)
            System_CLK : integer := 18432000    -- Системный CLK in Hz (default=50000000)
            );
    PORT (
        CLK          : in    Std_Logic;    -- сигнал тактовой частоты
        RXD           : in    Std_Logic;    -- eeeey Receive
        RST_N         : in    Std_Logic;    -- системный сброс
        DATA_IN      : in    Std_Logic_Vector(7 downto 0); -- данные для
передачи
        CHETN         : out   Std_Logic;    -- четность для передачи
        DATA_OUT     : out   Std_Logic_Vector(7 downto 0);    -- принятые
данные
        RX_VALID      : out   Std_Logic;    -- флаг "данные приняты"
        TX_VALID      : in    Std_Logic;    -- флаг "начать отправку"
        TXD           : out   Std_Logic;    -- линия TRANSMIT
        TX_BUSY       : out   Std_Logic;    -- флаг "передатчик занят"
        RX_BUSY       : out   Std_Logic;    -- флаг "приёмнки занят"
    );
end UART_1;
architecture BEHAV of UART_1 is
    signal TxBuf : Std_Logic_Vector(7 downto 0); -- буфер передатчика
    signal RxBuf : Std_Logic_Vector(7 downto 0); -- буфер приёмника
    signal prevRXD: Std_Logic;                -- RXD буфер регистра
    signal RxReady: Std_Logic;
    signal TXRead: Std_Logic;
    signal TxBitCnt : integer range 0 to 10;
    signal TxReady : Std_Logic;
    signal CntRX : integer range 0 to System_CLK/(UART_Speed);
    signal CntTX : integer range 0 to System_CLK/(UART_Speed);
    signal RxBitCnt: integer range 0 to 10;
    signal chet: std_logic;
begin
    UART_Tx: process(CLK, RST_N)
    begin
        if RST_N='0' then
            TXD<='1';
            TxBitCnt<=0;
            TxBuf<=(others=>'0');
            CntTX<= 0;
```

```

        TxReady <= '1';
    elsif (rising_edge(CLK)) then
        if (TX_VALID = '1' and TxReady = '1') then
            TxBuf(7 downto 0) <= DATA_IN(7 downto 0);
            TxReady <= '0';
            TxBitCnt <= 0;
            CntTX <= 0;
            chet <= DATA_IN(0) XOR DATA_IN(1) XOR
DATA_IN(2) XOR DATA_IN(3) XOR DATA_IN(4) XOR DATA_IN(5) XOR DATA_IN(6)
XOR DATA_IN(7);
        end if;
        if (TxReady = '0') then
            if CntTX = (System_CLK/(UART_Speed)) then
                CntTX <= 0;
                case TxBitCnt is
                    when 0 =>
                        TXD <= '0';      -- start bit
                        TxBitCnt <= TxBitCnt+1;
                    when 1|2|3|4|5|6|7|8 =>
                        TXD <= TxBuf(0);
                        TxBuf <= '0' & TxBuf(7 downto 1);
                        TxBitCnt <= TxBitCnt+1;
                    when 9 =>
                        TXD <= chet;
                        TxBitCnt <= TxBitCnt+1;
                        CHETN <= chet;
                    when 10 =>
                        TXD <= '1';  -- stop bit
                        TxBuf <= (others => '0');
                        TxBitCnt <= 0;
                        TxReady <= '1';
                end case;
            else
                CntTX <= CntTX+1;
            end if;
        end if;
    end if;
end process UART_Tx;
TX_BUSY <= not (TxReady);

UART_Rx: process(CLK, RST_N)
begin
    if RST_N = '0' then
        RxBitCnt <= 0;
        RxBuf <= (others => '0');
        RxReady <= '1';
        prevRXD <= '1';
        CntRX <= 0;
    elsif (rising_edge(CLK)) then
        if (RxReady = '1') then

```

```

        prevRXD <= RXD;
        if (RXD='0' and prevRXD='1') then          -- Start bit,
RxBitCnt<=0;                                     -- RX Bit counter
        RxReady <= '0';                          -- Start receiving
        RxBuf<=(others=>'0');
        CntRX <= 0;
    end if;
else
    if CntRX=(System_CLK/(UART_Speed*2)) then
        case RxBitCnt is
            when 0 =>
                if (RXD='1') then -- start bit failed
                    RxReady<='1';
                end if;
            when 1|2|3|4|5|6|7|8 =>
                RxBuf<= RXD & RxBuf(7 downto 1);
                RxReady<='0';

                when 9 =>
                    if (RXD XOR RxBuf(0) XOR RxBuf(1)
XOR RxBuf(2) XOR RxBuf(3) XOR RxBuf(4) XOR RxBuf(5) XOR RxBuf(6) XOR RxBuf(7))
= '1' then RxReady <= '0';
                    else
                        RxReady <= '1';
                    end if;
                when 10 =>
                    RxReady<='1';
                when others => RxReady<='0';

            end case;
            CntRX<=CntRX+1;
            RxBitCnt <= RxBitCnt+1;
            elsif (CntRX=(System_CLK/(UART_Speed))) then
                CntRX <= 0;
            else
                CntRX<=CntRX+1;
            end if;
        end if;
    end if;
end process UART_Rx;

DATA_OUT(7 downto 0)<= RxBuf(7 downto 0) when RxReady='1' else (others=>'0');
RX_VALID<=RxReady;
RX_BUSY<=not (RxReady);
end BEHAV;

```

ПРИЛОЖЕНИЕ В

(обязательное)

Код программы, реализующей кодер

ENTITY CODER1 IS

PORT

(

clk : in std_logic;

dan : in std_logic;

kd0, kd1, kd2, kd3, kd4, kd5, kd6, kd7, kd8, kd9, kd10, kd11, kd12, kd13, kd14,

kd15 :inout std_logic;

dan_out : out std_logic);

end coder1;

architecture kod of coder1 is

signal Sbit : integer range 0 to 67;

signal do : std_logic;

begin

process (clk)

begin

if (clk'event and clk='1') then

do<=dan;

if Sbit>=53 then

kd0<='0';

kd1<=kd0;

kd2<=kd1;

kd3<=kd2;

kd4<=kd3;

kd5<=kd4;

kd6<=kd5;

kd7<=kd6;

kd8<=kd7;

kd9<=kd8;

kd10<=kd9;

```

        kd11<=kd10;
        kd12<=kd11;
        kd13<=kd12;
        kd14<=kd13;
        kd15<=kd14;
        dan_out<=kd15;
    else
        kd0<=do XOR kd15;
        kd1<=kd0;
        kd2<=kd1;
        kd3<=kd2;
        kd4<=kd3;
        kd5<=kd4 XOR kd15;
        kd6<=kd5;
        kd7<=kd6;
        kd8<=kd7;
        kd9<=kd8;
        kd10<=kd9;
        kd11<=kd10;
        kd12<=kd11 XOR kd15;
        kd13<=kd12;
        kd14<=kd13;
        kd15<=kd14;
        dan_out<=kd15;
    end if;
    Sbit<=Sbit+1;
end if;
end process;
END;

```

ПРИЛОЖЕНИЕ Г

(обязательное)

Код программы, реализующей декодер

```
LIBRARY ieee;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
USE IEEE.numeric_std.ALL;
ENTITY DCD_V1 IS
    PORT
    (
        CLK          : IN STD_LOGIC;
        INFDM         : IN STD_LOGIC;
        TI1_1         : IN STD_LOGIC;
        TI1_2         : IN STD_LOGIC;
        TI5_1         : IN STD_LOGIC;
        TI8_4         : IN STD_LOGIC;
        NSB           : IN STD_LOGIC;
        PRKFP         : IN STD_LOGIC;
        regU1         : IN std_logic_vector(7 downto 0);--RR1PRM
        INT           : OUT integer range 0 to 70;
        PRM_69        : OUT STD_LOGIC;
        rez           : OUT std_logic
    );
END;
ARCHITECTURE RRR OF DCD_V1 IS
    signal    kd          : std_logic_vector(15 downto 0);--регистр декодера
    signal    SinPrm      : std_logic;--синхронизация приемника
    signal    Er          : std_logic;--ошибка
    signal    Sbitprm     : integer range 0 to 72; --счетчик бит приемника
    signal    chErr       : integer range 0 to 7; --счетчик ошибок
    signal    kd_o        : std_logic;
```

```

BEGIN
--счетчик битов приемника
process (CLK, NSB, regU1(0))
begin
if NSB='0' and regU1(0)='1' then
    if (CLK'event and CLK='1') then
        if PRKFP='1' then
            Sbitprm<=0;
            SinPrm<='1';
            chErr<=0;
            PRM_69 <='0';
        elsif TI1_1='1' and Sbitprm/=69 then
            PRM_69 <= '0';
            Sbitprm<=Sbitprm+1;
        elsif (TI1_1='1' and Sbitprm=69) then
            Sbitprm<=1;
            PRM_69 <='0';
        elsif TI8_4='1' and Sbitprm=69 then
            PRM_69 <='1';
            if kd="0000000000000000" then
                Er<='0';
                chErr<=0;
            else
                Er<='1';
                if chErr=4 then
                    SinPrm<='0';
                else
                    chErr<=chErr+1;
                end if;
            end if;
        end if;
    end if;
end if;
else
    chErr<=0;
    SinPrm<='0';

```

```

Er<='0';
Sbitprm<=0;
PRM_69 <='0';
end if;
end process;

--декодер
process (CLK, NSB, regU1(0))
begin
if NSB='0' and regU1(0)='1' then
    if (CLK'event and CLK='1') then
        if TI5_1='1' then
            if Sbitprm>=54 then
                kd(0)<= INFDM XOR kd(15);
                kd(5)<=kd(4) XOR (INFDM XOR kd(15));
                kd(12)<=kd(11) XOR (INFDM XOR kd(15));
            else
                kd(0)<=NOT (INFDM) XOR kd(15);
                kd(5)<=kd(4) XOR (NOT (INFDM) XOR kd(15));
                kd(12)<=kd(11) XOR (NOT (INFDM) XOR
kd(15));

            end if;

            kd(1)<=kd(0);
            kd(2)<=kd(1);
            kd(3)<=kd(2);
            kd(4)<=kd(3);
            kd(6)<=kd(5);
            kd(7)<=kd(6);
            kd(8)<=kd(7);
            kd(9)<=kd(8);
            kd(10)<=kd(9);
            kd(11)<=kd(10);
            kd(13)<=kd(12);
            kd(14)<=kd(13);
            kd(15)<=kd(14);

```

```
rez<=kd(15);  
end if;  
  
end if;  
else  
kd<="000000000000000000";  
end if;  
end process;  
END;
```

ПРИЛОЖЕНИЕ Д

(обязательное)

Код программы, реализующей адресацию (SK_1)

```
LIBRARY ieee;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
USE IEEE.numeric_std.ALL;
ENTITY SK_1 IS
    PORT
    (
        CLK                : IN STD_LOGIC;
        PUSK                : IN STD_LOGIC;
        TX_BUSY             : IN STD_LOGIC;
        DAN                 : IN STD_LOGIC_VECTOR (7 DOWNTO 0);
        ADR                 : OUT STD_LOGIC_VECTOR (7 DOWNTO 0);
        VALID               : OUT STD_LOGIC;
        RTS                 : OUT STD_LOGIC;
        SK                  : OUT STD_LOGIC;
        Z1                  : OUT integer range 0 to 31;
        CT1                 : OUT STD_LOGIC;
        ADRM1               : OUT STD_LOGIC_VECTOR (7 DOWNTO 0);
        R1                  : OUT STD_LOGIC;
        S1                  : OUT STD_LOGIC_VECTOR (3 downto 0);
        CCC1                : OUT STD_LOGIC_VECTOR (7 downto 0);
        SK1                 : OUT STD_LOGIC:='0';
        SK2                 : OUT STD_LOGIC:='0';
        SK3                 : OUT STD_LOGIC:='0';
        SK4                 : OUT STD_LOGIC:='0'
    );
END;
ARCHITECTURE b OF SK_1 IS
    signal      reg                : std_logic_vector(7 downto 0);
    signal      s                  : std_logic_vector(3 downto 0);
    signal      z                  : integer range 0 to 31 ;
    signal      cn,cn1             : integer range 0 to 25000000;
    signal      const, const1      : integer range 0 to 25000000;
    signal      adrm, adrr, ccc     : std_logic_vector(7 downto 0);
    signal      r, p, ct, a, b, sk_vn : std_logic;
    signal      zad                 : integer range 0 to 511;
    signal      pu, vkl, t, ru      : std_logic:='0';
BEGIN
    process (CLK)
    begin
        if (CLK'event and CLK='1') then
```

```

If PUSK='1' then
    sk_vn <= '1';
    if cn=18433 and z=0 then
        adrm<="00000000";-- PRD odk1 VKL
        p<='1';
        ct<='1';
        const<=88593;
        cn<=cn+1;
        z<=z+1;
    elsif cn=const and z=1 then
        adrm<="00001000";-- PRM1 odk1 VKL
        p<='1';
        ct<='1';
        cn<=cn+1;
        z<=z+1;
        const<=const+150000;
    elsif cn=const and z=2 then
        adrm<="00010000";-- kdg
        p<='1';
        ct<='1';
        cn<=cn+1;
        z<=z+1;
        SK1<='1';
        const<=const+650000;
    elsif cn=const and z=3 then
        adrm<="01000000";
        p<='1';
        ct<='1';
        cn<=cn+1;
        z<=z+1;
        const<=const+15000;
    elsif cn=const and z=4 then
        adrm<="01001000";
        p<='1';
        ct<='1';
        cn<=cn+1;
        z<=z+1;
        const<=const+15000;
    elsif cn=const and z=5 then
        adrm<="00100000";
        p<='1';
        ct<='1';
        cn<=cn+1;
        z<=z+1;
        const<=const+15000;
    elsif cn=const and z=6 then
        adrm<="00101000";-- PRM1 odk2
        p<='1';
        ct<='1';
        cn<=cn+1;

```

```

        z<=z+1;
        const<=const+15000;
    elsif cn=const and z=7 then
        adrm<="00010000";-- kdg
        p<='1';
        ct<='1';
        cn<=cn+1;
        z<=z+1;
        SK2<='1';
        const<=const+700000;
    elsif cn=const and z=8 then
        adrm<="01000000";-- PRD VYKL
        p<='1';
        ct<='1';
        cn<=cn+1;
        z<=z+1;
        const<=const+15000;
    elsif cn=const and z=9 then
        adrm<="01001000";-- PRM VYKL
        p<='1';
        ct<='1';
        cn<=cn+1;
        z<=z+1;
        const<=const+15000;
    elsif cn=const and z=10 then
        adrm<="00110000";-- PRD odk2-1 VKL
        p<='1';
        ct<='1';
        cn<=cn+1;
        z<=z+1;
        const<=const+15000;
    elsif cn=const and z=11 then
        adrm<="00111000";-- PRM odk2-1 VKL
        p<='1';
        ct<='1';
        cn<=cn+1;
        z<=z+1;
        const<=const+15000;
    elsif cn=const and z=12 then
        adrm<="00010000";-- kdg
        p<='1';
        ct<='1';
        cn<=cn+1;
        z<=z+1;
        SK3<='1';
        const<=const+700000;
    elsif cn=const and z=13 then
        adrm<="01000000";-- PRD VYKL
        p<='1';
        ct<='1';

```

```

cn<=cn+1;
z<=z+1;
const<=const+15000;
elsif cn=const and z=14 then
adrm<="01001000";-- PRM VYKL
p<='1';
ct<='1';
cn<=cn+1;
z<=z+1;
const<=const+15000;
elsif cn=const and z=15 then
adrm<="00000000";-- PRd1 odk-1 bezadres
p<='1';
ct<='1';
cn<=cn+1;
z<=z+1;
const<=const+15000;
elsif cn=const and z=16 then
adrm<="01010000";-- PRM odk-1 bezadres
p<='1';
ct<='1';
cn<=cn+1;
z<=z+1;
const<=const+15000;
elsif cn=const and z=17 then
adrm<="01011000";-- KDG
p<='1';
ct<='1';
cn<=cn+1;
z<=z+1;
SK4<='1';
const<=const+5000000;
elsif cn=const and z=18 then
adrm<="01000000";-- PRD VYKL
p<='1';
ct<='1';
cn<=cn+1;
z<=z+1;
const<=const+15000;
elsif cn=const and z=19 then
adrm<="01001000";-- PRM VYKL
p<='1';
ct<='1';
cn<=cn+1;
z<=z+1;
const<=const+15000;
elsif cn=const and z=20 then
sk_vn <= '0';
SK1<='0';
SK2<='0';

```

```

        SK3<='0';
        SK4<='0';
    elsif cn/=25000000 then--580000
        cn<=cn+1;
    if ru='1' then
        p<='0';
        ct<='0';
    end if;
    end if;
else
    cn<=0;
    z<=0;
    const<=0;
end if;
end if;
end process;
CT1<=ct;
R1<=r;
RTS<=ct or r;
SK <= sk_vn;
process (CLK)
begin
    if (CLK'event and CLK='1') then
        if p='1' then
            if r='0' then--and b='0'
                if (TX_BUSY = '0') or a = '1' then
                    if s="0000" then
                        s<=s+1;
                        a<='1';
                        adrr<=adrm+ccc;
                    elsif s="0001" then
                        s<=s+1;
                        if DAN="11111111" then
                            r<='1';
                            a<='0';
                            b<='1';
                            ccc<="00000000";
                            s<="0000";
                            adrr<="00000000";
                        else
                            VALID<='1';
                        end if;
                    elsif s="0011" then
                        s<=s+1;
                        VALID<='0';
                        ccc<=ccc+1;
                    elsif s="0101" then
                        s<="0000";
                        a<='0';
                    else

```

```

                                s<=s+1;
                                end if;
                                end if;
                                else
                                    if zad=100 then
                                        r<='0';
                                        ru<='1';
                                        zad<=0;
                                    else
                                        zad<=zad+1;
                                    end if;
                                end if;
                                else
                                    ru <= '0';
                                    b <='0';
                                    ccc<="00000000";
                                end if;
                                end if;
                                end process;
                                ADRM1<=adrm;
                                ADR<=adrr;
                                Z1<=z;
                                CCC1<=ccc;
                                S1<=s;
                                END;

```

ПРИЛОЖЕНИЕ Е

(обязательное)

Код программы, реализующей проверку правильности принятия
сообщения

```
LIBRARY ieee;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
USE IEEE.numeric_std.ALL;
ENTITY DCD_V1_1 IS
    PORT
    (
        CLK          : IN STD_LOGIC;
        NSB           :IN STD_LOGIC;
        RES_SK1       : OUT STD_LOGIC:= '0';
        RES_SK2       : OUT STD_LOGIC:= '0';
        RES_SK3       : OUT STD_LOGIC:= '0';
        RES_SK4       : OUT STD_LOGIC:= '0';
        ZAPRc         :OUT STD_LOGIC;
        SK1           :IN STD_LOGIC;
        SK2           : IN STD_LOGIC;
        SK3           : IN STD_LOGIC;
        SK4           : IN STD_LOGIC
    );
END;
ARCHITECTURE RRR OF DCD_V1_1 IS
    signal      zaprPrm, zaprPrm1  :std_logic; --запросы
BEGIN
    process (CLK, NSB)
    begin
        if NSB='0' then
            if (CLK'event and CLK='1') then
                if zaprPrm='1'and SK='1'and SK1='1' then
                    RES_SK1<='1';
                    zaprPrm1<='0';
                end if;
                if zaprPrm='1'and SK='1'and SK2='1' then
                    RES_SK2<='1';
                    zaprPrm1<='0';
                end if;
                if zaprPrm='1'and SK='1'and SK3='1' then
                    RES_SK3<='1';
                    zaprPrm1<='0';
                end if;
                if zaprPrm='1'and SK='1'and SK4='1' then
                    RES_SK4<='1';
                end if;
            end if;
        end if;
    end process;
```

```
        zaprPrm1<='0';
    end if;
    if zaprPrm='1' and SK='0' then
        zaprPrm1<='1';
    end if;
    if zaprPrm<='0' and SK='0' then
        zaprPrm1<='0';
    end if;
end if;
end process;
END;
```

ПРИЛОЖЕНИЕ Ж

(обязательное)

Протокол проверки ВКР на оригинальность

Протокол

проверки на оригинальность в системе «Антиплагиат.ВУЗ» выпускной квалификационной работы на тему «Узлы обеспечения целостности данных в аппаратуре передачи данных специального назначения» студентки шестого курса О.М.Подгорецкой специальности 10.05.02 «Информационная безопасность телекоммуникационных систем».

Руководитель ВКР Бессонов А.В.

Информация о документе:

Имя исходного файла: ВКР_11ПТ1_ПодгорецкаяОМ.docx

Имя компании: Пензенский гос. Университет

Тип документа: Прочее

Имя документа: Группа 11ПТ1 ВКР Подгорецкая ОМ

Дата проверки: 06.01.2017 18:11

Модули поиска: Кольцо вузов, Интернет (Антиплагиат), Модуль поиска ЭБС "Лань", Модуль поиска ЭБС БиблиоРоссика, Диссертации и авторефераты РГБ, Университетская библиотека онлайн, Пензенский гос. Университет, Цитирования

Текстовые

статистики:

Индекс читаемости: сложный

Неизвестные слова: в пределах нормы

Макс. длина слова: в пределах нормы

Большие слова: в пределах нормы

Оригинальные блоки: 84,34%

Заимствованные блоки: 15,54%

Заимствование из "белых" источников: 0,13%

Итоговая оценка оригинальности: **84,46%**

7.01.2017

_____ Бессонов А.В.