

Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых

На правах рукописи



Позднякова Лидия Васильевна

**РАЗВИТИЕ МЕТОДОВ КОРРЕКЦИИ КОМПЛЕКСНОЙ
ПЕРЕДАТОЧНОЙ ХАРАКТЕРИСТИКИ В СИСТЕМАХ С
ОРТОГОНАЛЬНЫМ ЧАСТОТНЫМ РАЗДЕЛЕНИЕМ КАНАЛОВ И
МУЛЬТИПЛЕКСИРОВАНИЕМ (OFDM)**

05.12.04 – Радиотехника, в том числе системы
и устройства телевидения

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель: д.т.н, профессор,
заслуженный деятель науки РФ Никитин О. Р.

Владимир – 2016

Содержание

Введение	5
Глава 1. Аналитический обзор методов коррекции комплексной передаточной характеристики системы связи на основе ортогонального частотного разделения каналов с мультиплексированием	13
1.1. Технология ортогонального частотного разделения каналов с мультиплексированием.....	13
1.2. Синхронизация приёмо-передающих устройств в OFDM системе.....	25
1.3. Обзор методов коррекции комплексной передаточной характеристики	37
1.4. Применение пилот-поднесущих для коррекции КПХ системы связи	45
Выводы по главе 1.....	47
Глава 2. Развитие методов синхронизации и коррекции комплексной передаточной характеристики системы связи по преамбуле и пилот-поднесущим	49
2.1. Алгоритм коррекции КПХ системы связи с применением в расчетах полярной системы координат.....	49
2.2. Алгоритм коррекции КПХ системы связи с использованием в расчетах ортогональной системы координат	55
2.3. Целочисленный алгоритм коррекции КПХ системы связи с использованием в расчетах ортогональной системы координат.....	59
2.4. Определение оптимальной преамбулы для предложенного алгоритма коррекции КПХ системы связи	65
2.5. Способ временной синхронизации для преамбулы, имеющей дискретно-непрерывный набор спектральных компонент.....	73
2.6. Тактовая и частотная синхронизация в канале связи OFDM.....	86
Выводы по главе 2.....	89
Глава 3. Анализ возможностей разработанных алгоритмов методами математического моделирования и апробации	90
3.1. Алгоритм генерации шума в заданной полосе частот.....	90
3.2. Программа моделирования коррекции КПХ	93
3.3. Моделирование алгоритма коррекции КПХ, использующего в расчетах полярную систему координат.....	96
3.4. Моделирование предложенного алгоритма коррекции КПХ	100

3.5. Сравнение алгоритмов и определение минимального отношения сигнал/шум с учетом помехоустойчивого кодирования для модуляций QAM16, QAM64 и преамбулы состоящей из нескольких символов.....	104
3.6. Оценка точности предложенного целочисленного алгоритма коррекции КПХ по сравнению с вещественным алгоритмом.....	111
3.7. Моделирование способа временной синхронизации для преамбулы, имеющей дискретно-непрерывный набор спектральных компонент	113
Выводы по главе 3.....	115
Заключение.....	117
Литература.....	120
П.1. Алгоритмы работы программы моделирования	129
П.2. Моделирование модифицированного алгоритма коррекции КПХ канала связи с уменьшенным СКЗ	159
П.3. Моделирование алгоритма коррекции КПХ, использующего в расчетах полярную систему координат с увеличенными по уровню значениями пилот-поднесущих	161
П.4. Моделирование модифицированного алгоритма коррекции КПХ канала связи при различных значениях параметра «сдвиг от начала символа».....	163
П.5. Заявка на изобретение 2015109441 «Способ временной синхронизации системы связи на основе ортогонального частотного разделения каналов с мультиплексированием по преамбуле»	165
П.7. Свидетельство государственной регистрации программы для ЭВМ RU 2015618752	167
П.8. Свидетельство государственной регистрации программы для ЭВМ RU 2015618979	168
П.9. АКТ использования научных результатов диссертации в ООО «Предприятие по модернизации авиационных комплексов»	169
П.10. АКТ внедрения научных результатов диссертации в ФГБОУ «ВлГУ»	170

Основные сокращения

HDL (англ. hardware description language) – язык описания аппаратуры.

OFDM (англ. Orthogonal frequency division multiplexing) — ортогональное частотное разделение каналов с мультиплексированием.

АБГШ – аддитивный белый Гауссовский шум.

АС – абонентская станция.

АЦП – аналого-цифровой преобразователь.

АЧХ – амплитудно-частотная характеристика.

БПФ – быстрое преобразование Фурье.

БС – базовая станция.

ГВЗ – групповое время запаздывания.

ККП – комплексный коэффициент передачи.

КПХ – комплексная передаточная характеристика.

МСИ – межсимвольная интерференция.

ОБПФ – обратное быстрое преобразование Фурье.

ОККП – обратный комплексный коэффициент передачи.

ОКПХ – обратная комплексная передаточная характеристика.

ПЛИС – программируемая логическая интегральная схема.

СКЗ – среднеквадратическое значение.

СКО – среднеквадратическое отклонение.

ФЧХ – фазо-частотная характеристика.

ЦОС – цифровая обработка сигналов.

Введение

Актуальность темы

Ортогональное частотное разделение каналов с мультиплексированием (*OFDM*) известно давно, но оборудование, реализующее данный принцип работы (*WiFi*, *WiMAX*, *LTE*), стало доступным сравнительно недавно, и представлено в основном крупными производителями, такими как *Intel*, *Fujitsu*, *Huawei*. Большая задержка между появлением самой технологии и появлением оборудования возникла вследствие того, что быстродействие цифровых устройств было недостаточным для реализации приемлемой скорости передачи данных и только в последнее время вышло на требуемый уровень. Тем не менее, реализовать требуемую скорость только на одном процессоре без применения специализированных микросхем, рассчитывающих самые критические по скорости процедуры, такие как преобразование Фурье, эквалайзер и помехоустойчивое кодирование, не представляется возможным. Отечественные разработки цифровых *OFDM* модемов (необходимых для организации беспроводной высокоскоростной сети передачи данных) базируются на применении зарубежных микросхем, реализующих цифровую обработку, такую как модуляция/демодуляция и кодирование/декодирование. Однако внутренняя структура этих микросхем закрыта и нет возможности не только внести изменения и усовершенствования в алгоритмы работы, но и проверить их на отсутствие ошибок и «недокументированных возможностей», что может представлять угрозу безопасности. Отечественные разработки представляют собой сборку из «готовых» блоков, закрытых для проверки и модернизации. Такой подход не позволяет быть уверенным в надежности устройства и не развивает отечественную прикладную науку. Актуальным является создание отечественного *OFDM* модема, который будет обеспечивать все этапы обработки данных в передающем (шифрование, рандомизация, кодирование, внесение избыточности, скремблирование, модуляция) и приемном (учет

комплексного коэффициента передачи, демодуляция, дескремблирование, декодирование, дерандомизация, дешифрование) трактах.

Важной задачей создания отечественной *OFDM* системы является разработка эффективного алгоритма коррекции комплексной передаточной характеристики канала связи, как необходимой составной части реализации физического сетевого уровня. [1] Для этого с наименьшими аппаратными затратами, высоким качеством, точностью и достоверностью получаемой информации могут быть использованы современные быстродействующие микроконтроллеры (МК) и программируемые логические интегральные схемы (ПЛИС).

Совершенствуются технологии цифро-аналогового и аналого-цифрового преобразования, позволяя работать с цифровыми сигналами высокой разрядности, с минимальными искажениями и погрешностями в режиме реального времени. Всё большее число задач решается не на аппаратном, а на алгоритмическом и программном уровне. Использование встроенных вычислительных средств обеспечивает гибкость, надёжность, низкую стоимость и высокое быстродействие цифровых радиоустройств и радиосистем.

Как показывает практика, при передаче широкополосных сигналов часто возникает необходимость коррекции влияния комплексной передаточной характеристики системы связи на принятый сигнал. В зависимости от свойств стационарности канала частотная характеристика может корректироваться в начале сеанса работы, периодически или в реальном времени. Современные алгоритмические подходы позволяют быстрее и с меньшими затратами проводить оценку частотных характеристик в реальном времени, что даёт возможность использования этих данных в цифровых модемах для адаптивной корректировки влияния радиоканала.

Большой вклад в разработку методов и создание аппаратуры для оценки частотных характеристик внесли творческие коллективы ряда российских высших учебных заведений и научно-исследовательских

институтов. Активно в данной области работает ряд зарубежных фирм, таких как *Fujitsu, Intel, Huawei, National Instruments* и др. Вопросам цифровой обработки посвящены работы известных зарубежных и отечественных учёных, среди которых: Андрака Р., Байков В.Д., Вишневский В.М., Гантмахер В.Е., Котельников В.А., Кнут Д., Крухмалёв В.В., Меерсон А.М., Раушер К., Рашич А.В. и многие др. В настоящее время опубликованы сотни работ, освещающие различные теоретические и практические вопросы цифровой обработки сигналов (ЦОС). [2]

Анализ публикаций по применению методов ЦОС в радиоэлектронной аппаратуре (РЭА) показывает, что задачам реализации алгоритмов оценки параметров радиоканалов в реальном времени уделено недостаточно внимания. Вместе с тем, в связи с быстрым развитием средств вычислительной техники методология, алгоритмизация и программное обеспечение обработки дискретизированных сигналов отстают в своём развитии от возможностей современных встраиваемых в РЭА одноплатных компьютеров, МК и ПЛИС.

Таким образом, в настоящее время существует актуальная техническая и научная задача разработки и создания эффективных алгоритмических средств для корректировки влияния комплексной передаточной характеристики системы связи на принятый сигнал.

Цель исследования: разработка и моделирование алгоритма коррекции комплексной передаточной характеристики, обеспечивающего снижение требований к ресурсам, необходимым для его реализации при поддержании временной и тактовой синхронизации.

Основными задачами диссертационной работы являются:

1. Разработка быстродействующего алгоритма коррекции комплексной передаточной характеристики (КПХ) на основе преамбулы и пилот-поднесущих.

2. Разработка алгоритмов тактовой и временной синхронизации при использовании преамбулы, содержащей дискретно-непрерывный набор

частотных компонент, т.е. не имеющей во временной области повторяющихся участков.

3. Создание на языке высокого уровня программ моделирования для обоснованного выбора рабочих параметров алгоритмов тактовой и временной синхронизации, а также алгоритма коррекции комплексной передаточной характеристики.

Объектом исследования являются методы коррекции передаточной характеристики, имеющие возможность реализации, во встроенных в современную радиоэлектронную аппаратуру системах ЦОС.

Предметом исследования являются алгоритмы цифровой обработки данных для корректировки влияния комплексной передаточной характеристики на принятый сигнал, а также методики временной и частотной синхронизации, которые необходимы для создания рабочих условий применения алгоритма коррекции.

Методы исследований

В работе использованы методы математической статистики, спектрального анализа и математического моделирования, реализованного на компьютерном языке высокого уровня и языке описания аппаратуры.

Моделирование и проектирование с использованием стандартных программ для проектирования и симуляции, а также разработка специальных компьютерных программ моделирования. Апробация на имеющемся на кафедре РТ и РС оборудовании, а именно перенос разработанного алгоритма на язык описания аппаратуры *Verilog* для тестирования алгоритма коррекции КПХ в условиях реального высокочастотного канала связи. Отладочная плата *OFDM* модема имеет следующие параметры:

1. Микроконтроллер 32х битный *AVR AP7000*, 160 МГц.
2. ОЗУ 64 МБ, *SDRAM*.
3. ПЗУ 2 ГБ, *SDCARD*.
4. ПЛИС *Altera Cyclone 3 EP3C120* (120000 ЛЭ).
5. Радиомодуль 3.4-3.6 ГГц, полоса 3.5, 5, 5.5, 7 МГц.

Используя имеющиеся отладочные платы *OFDM* модемов, одна из которых представлена на рисунке 1 можно будет оценить параметры разработанного алгоритма, такие как занимаемая емкость, в логических элементах, максимальная скорость работы и провести его оптимизацию.

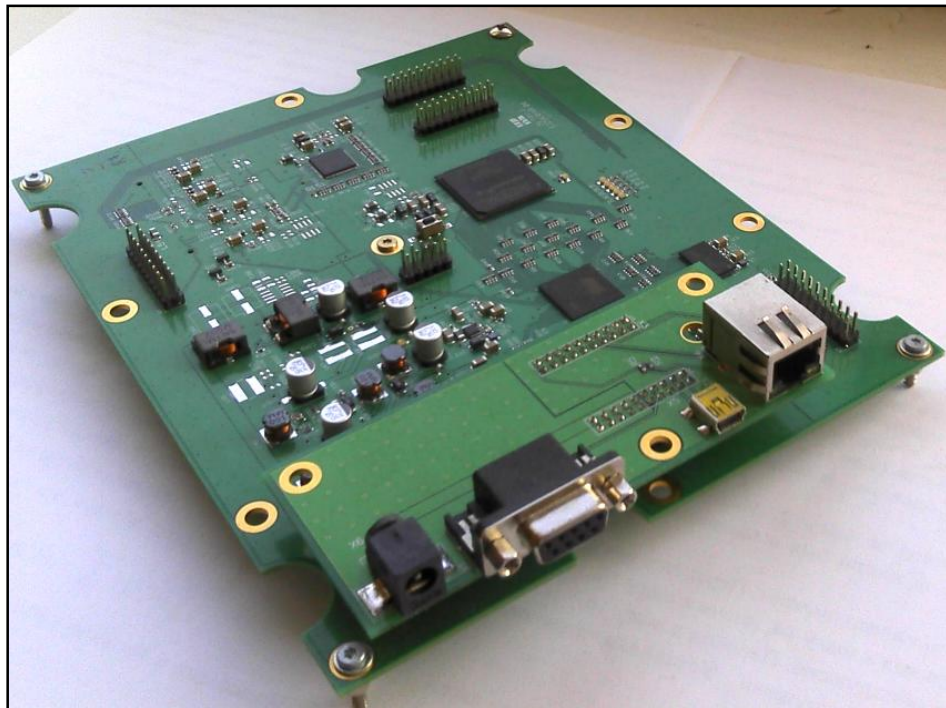


Рисунок 1. *OFDM* модем с установленным радиомодулем

Научная новизна работы заключается в развитии метода коррекции комплексной передаточной характеристики, также в создании методики временной, тактовой и частотной синхронизации, необходимой для функционирования предложенного метода коррекции:

1. Разработан быстродействующий алгоритм компенсации влияния комплексной передаточной характеристики (КПХ) на принятый сигнал на основе преамбулы и пилот-поднесущих. Обоснован выбор уровня пилот-поднесущих по отношению к уровню данных. Даны рекомендации по использованию разделения преамбулы на символы.

2. Разработаны алгоритмы временной, тактовой и частотной синхронизации при использовании преамбулы содержащей один, два или более двух символов, каждый из которых во временной области не имеет

повторяющихся участков, а в частотной области содержит дискретно-непрерывный набор спектральных компонент.

3. Показано, что неравномерность группового времени запаздывания в пределах длины циклического префикса не влияет на работоспособность предложенных алгоритмов.

Практические результаты диссертации были достигнуты в процессе выполнения научно-исследовательских работ по договору о предоставлении гранта Фондом содействия развитию малых форм предприятий в научно-технической сфере. Перечень результатов, имеющих практическую ценность:

1. Применение разработанного алгоритма компенсации влияния комплексной передаточной характеристики на принятый сигнал на основе преамбулы и пилот-поднесущих позволило уменьшить количество занимаемых логических элементов на ПЛИС с 42400 до 38120 за счет исключения *CORDIC* модулей перевода из ортогональной системы координат в полярную и из полярной системы координат в ортогональную.

2. Использование повышенного уровня пилот-поднесущих по сравнению с уровнем сигнала позволило на 1,3 дБ снизить требования к отношению сигнал/шум по отношению к стандартному алгоритму.

3. Созданы программы моделирования коррекции КПХ, которые:

- формируют преамбулу и символ с данными;
- рассчитывают по преамбуле обратную КПХ;
- применяют к символу данных обратную КПХ;
- корректируют по пилот-поднесущим обратную КПХ;
- применяют к символу данных скорректированную по пилот-поднесущим обратную КПХ.

4. Создана программа моделирования алгоритма временной синхронизации, которая рассчитывает дисперсию оценки временного положения преамбулы от идеального значения в зависимости от отношения сигнал/шум для АБГШ канала. Для преамбулы имеющей полезную

длительность 1024 отсчета дисперсия равна нулю при отношении сигнал/шум -6 дБ для выборки из 100 000 реализаций.

Результаты работы

Работа обобщает теоретическую проработку и практической опыт реализации цифровых методов коррекции комплексных передаточных характеристик. Практические результаты работы использовались в проводимых на кафедре радиотехники и радиосистем Владимирского государственного университета научно-исследовательских работах в период с 2012 по 2015 гг.

Апробация работы

По материалам диссертации автором сделано 2 доклада на научной конференции «Перспективные технологии в средствах передачи информации ПТСПИ-2015» (Владимир-Суздаль, 2015). Подана заявка на патент на изобретение: «Способ временной синхронизации системы связи на основе ортогонального частотного разделения каналов с мультиплексированием по преамбуле» и получено положительное решение о выдаче патента РФ от 25.10.2016 (Приложение П.5). Получено 3 авторских свидетельства на регистрацию программ для ЭВМ. (Приложения П.6-8)

Внедрение

Внедрение теоретических и практических результатов работы произведено на ООО «Предприятие по модернизации авиационных комплексов». (Приложение П.9) Программные средства на основе разработанных алгоритмов применяются в учебном процессе на кафедре радиотехники и радиосистем (РТ и РС) ВлГУ. (Приложение П.10)

Проведены испытания OFDM модемов, в основе которых реализованы алгоритмы временной, тактовой синхронизации и коррекции комплексной передаточной характеристики.

Публикации по работе

По тематике исследований опубликовано 14 работ, из которых 6 в журналах из перечня рекомендованных ВАК.

На защиту выносятся научно обоснованные технические разработки, имеющие существенное значение для экономики страны, в рамках решения задачи развития методов компенсации влияния комплексной передаточной характеристики на принятый сигнал на основе преамбулы и пилот-поднесущих.

1. Методика:

- Тактовой и частотной синхронизации за счет применения алгоритма временной синхронизации.

2. Алгоритмы:

- Компенсации влияния комплексной передаточной характеристики радиоканала на принятый сигнал.
- Временной синхронизации при использовании преамбулы, не имеющей во временной области повторяющихся участков, т.е. содержащей дискретно-непрерывный набор частотных компонент.

3. Программы моделирования:

- Алгоритма коррекции комплексной передаточной характеристики радиоканала в условиях аддитивного белого Гауссова шума.
- Алгоритма временной синхронизации при использовании преамбулы содержащей дискретно-непрерывный набор частотных компонент.

Структура работы

Диссертация состоит из введения, трех глав, заключения, приложений, списка литературы, имеющего 78 наименований отечественных и зарубежных источников, в том числе 14 работ автора. Общий объем диссертации 170 страниц, в том числе, 119 страниц основного текста, 9 страниц списка литературы, 62 рисунка, 8 таблиц, 42 страницы приложений.

Глава 1. Аналитический обзор методов коррекции комплексной передаточной характеристики системы связи на основе ортогонального частотного разделения каналов с мультиплексированием

Задачей главы 1 является исследование задач, методов и проблем синхронизации, а также коррекции КПХ системы связи на основе ортогонального частотного разделения каналов с мультиплексированием (*OFDM*).

В параграфе 1.1 рассмотрены основные преимущества технологии *OFDM*. Показано, что введение циклического префикса позволяет эффективно бороться с межсимвольной интерференцией.

В параграфе 1.2 рассмотрены задачи и методы синхронизации *OFDM* – устройств систем, а именно временная, тактовая и частотная синхронизации.

В параграфе 1.3 приведены варианты построения эквалайзеров. Рассмотрены одномерные, двумерные и комбинированные эквалайзеры.

В параграфе 1.4 рассмотрено применение пилот-поднесущих для непрерывной подстройки КПХ.

1.1. Технология ортогонального частотного разделения каналов с мультиплексированием

Известны цифровые виды модуляции во временной области, такие как *BPSK*, *QPSK*, *QAM*. [3,4,5]

Однако из-за проблемы межсимвольной интерференции, особенно актуальной на высоких скоростях, модуляция во времени начинает использоваться все реже.

Межсимвольная интерференция является одной из разновидностей помех характерных для цифрового канала, она обусловлена воздействием одного символа на другой. [6] Эффект наблюдается как в проводных, так и в беспроводных системах передачи данных. [7]

Кабельная линия представляет собой распределенную *RC*-цепочку, и высокочастотные компоненты сигнала в ней подвержены затуханию.

При передаче данных по радиоканалу нет *RC* составляющей, но включается механизм многолучевого распространения, приводящий к подобному эффекту. Сигнал от передатчика достигает приемника по нескольким разным путям. Причины этого - отражения (например, от зданий), рефракция (преломление при прохождении через кроны деревьев) и атмосферные эффекты. Так как все пути разной длины, то они приводят к разным задержкам сигнала, в результате разные версии сигнала придут к приемнику в разное время. Из-за взаимного наложения всех этих сигналов результирующий сигнал будет искажен.

Таким образом, если передавать данные сигнальных созвездий на высокой скорости во временной области, встает проблема межсимвольной интерференции. Решить эту проблему позволяет технология *OFDM*. [8]

Модуляция *OFDM* (англ. *Orthogonal frequency-division multiplexing* — ортогональное частотное разделение каналов с мультиплексированием) является цифровой схемой модуляции, которая использует большое количество близко расположенных ортогональных поднесущих. [9,10] Каждая поднесущая модулируется по обычной схеме модуляции (например, квадратурная амплитудная модуляция) на низкой символьной скорости, сохраняя общую скорость передачи данных, как и у обычных схем модуляции одной несущей в той же полосе пропускания. [11, 12] На практике сигналы *OFDM* получаются путем использования БПФ (Быстрое преобразование Фурье), которое широко используется в радиотехнике. [13]

Основным преимуществом *OFDM* по сравнению со схемой с одной несущей является её способность противостоять сложным условиям в канале.[14] Например, бороться с затуханием в области ВЧ в длинных медных проводниках, с узкополосными помехами и частотно-избирательным затуханием, вызванным многолучевым характером распространения [15], без использования сложных аналоговых фильтров-эквалайзеров.

Канальная эквализация упрощается вследствие того, что *OFDM* сигнал может рассматриваться как множество медленно модулируемых узкополосных сигналов, а не как один быстро модулируемый широкополосный сигнал. Низкая символьная скорость делает возможным использование защитного интервала между символами, что позволяет справляться с временным рассеянием и устранять межсимвольные искажения (МСИ).

В технологии *OFDM* частотный диапазон разбивается равномерно между поднесущими (дополнительные несущие), количество которых может достигать до нескольких тысяч. Каждому передаваемому потоку назначается несколько таких поднесущих, т.е. каждый поток разбивается на N поднесущих. Каждая поднесущая содержит в себе сигнальное созвездие, модулированное соответствующим видом модуляции. Поднесущие между собой ортогональны. Эта особенность определяет многие положительные качества техники *OFDM*.

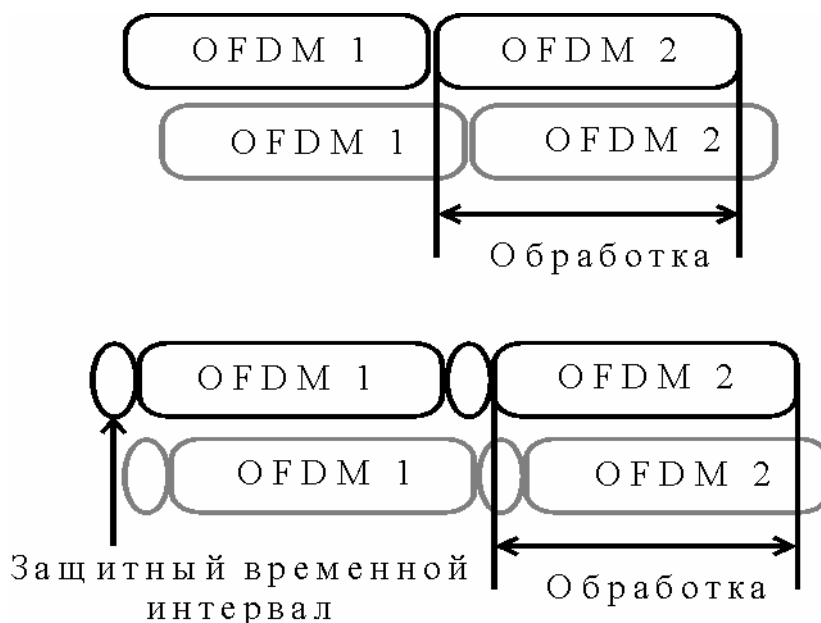


Рисунок 1.1.1 - Использование защитного интервала

Для борьбы с многолучевым распространением в *OFDM* [16] включён защитный интервал. Это возможно сделать, т.к. быстрый поток данных делится между поднесущими, на каждой из которых скорость подпотока

меньше первоначальной. За счёт этого можно выделить отрезок времени, который будет защищать основной сигнал от многолучевого распространения. Длительность этого защитного интервала может составлять $1/4$, $1/8$, $1/16$ или $1/32$ от длительности *OFDM* символа. На рисунке 1.1.1 поясняется принцип использования временного защитного интервала.

Обычно в качестве защитного интервала используют так называемый циклический префикс, являющийся копией окончания сигнала размещённой впереди. Это позволяет сохранить ортогональность. Чем дольше защитный интервал, тем в более сложных условиях может передаваться *OFDM* сигнал. [17]

Ортогональность поднесущих позволяет системам хорошо справляться с узкополосными помехами, которые могут подавить часть поднесущих. Благодаря корректирующим кодам информацию можно извлечь из неповреждённых поднесущих. Помимо этого, в *OFDM* каждая поднесущая может модулироваться различной схемой модуляции, например, *QPSK*, *16-QAM* или *64-QAM*. Как отмечалось выше, в таком подходе можно адаптивно регулировать помехоустойчивость и скорость потока данных для каждого канала (пользователя) в отдельности. [18]

Технической реализации *OFDM* не было долгое время, поскольку решение задачи аналоговыми методами весьма проблематично. С появлением быстрых вычислительных систем задача была реализована с помощью цифровых методов обработки сигналов. В основе подхода лежит преобразование Фурье, а точнее алгоритм быстрого преобразования Фурье. Синтетическим методом создаётся спектр сигнала, из которого обратным быстрым преобразованием Фурье (ОБПФ) получается аналоговый сигнал. Спектр такого сигнала уже состоит из ортогональных поднесущих, этот факт получается по определению преобразования Фурье. На рисунке 1.1.2 схематично показана архитектура типового приёмника и передатчика *OFDM*.

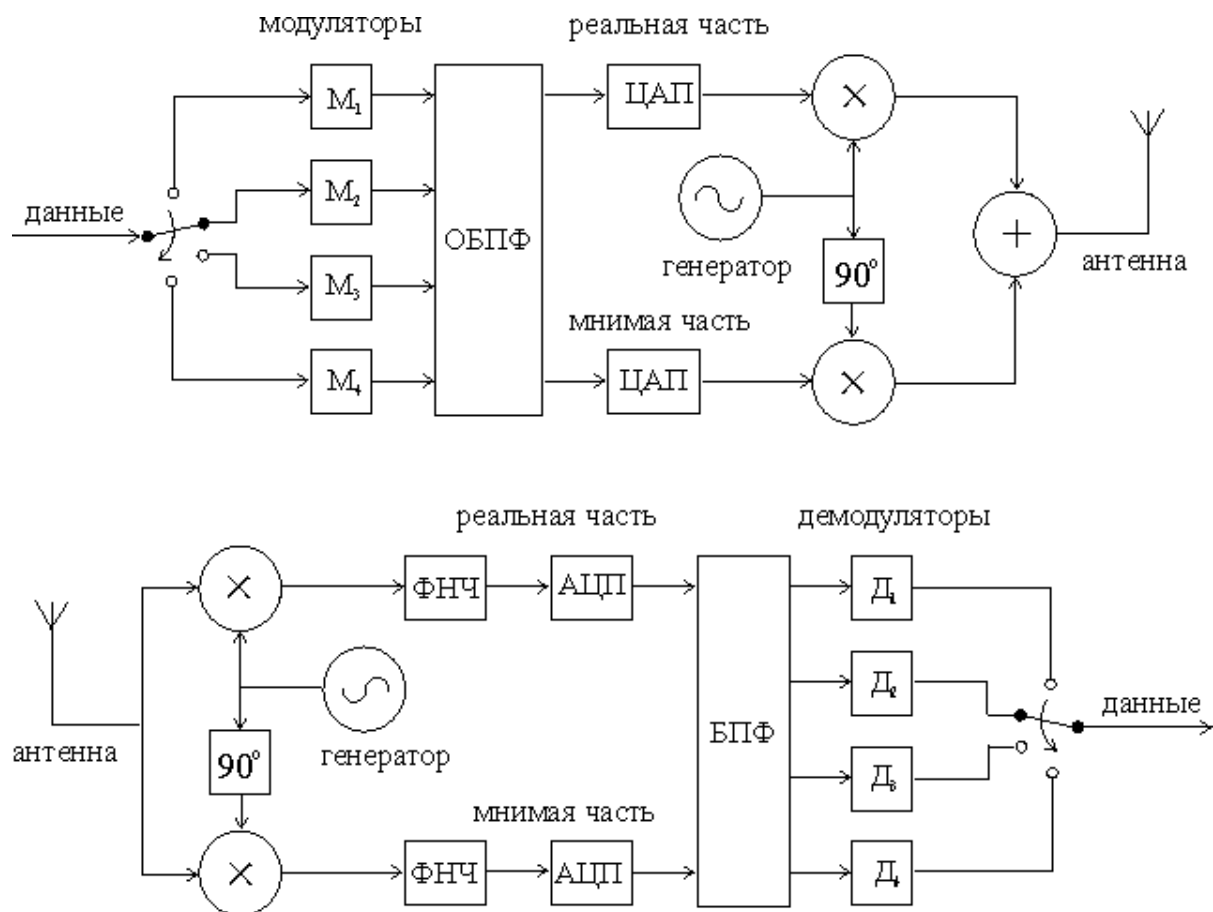


Рисунок 1.1.2 - Архитектура типового приёмника и передатчика *OFDM*

Непосредственное формирование сигнала после цифрового синтеза, который затем передаётся в антенну для излучения, происходит аналогично схеме *QAM* модуляции. В отдельности формируются квадратурные сигналы как мнимая и реальная часть синтезируемого сложного сигнала, а затем происходит его «сборка» и передача в антенну.

В виду того, что алгоритм БПФ/ОБПФ работает эффективно с выборками размерности кратными степеням двойки, то количество поднесущих в *OFDM* используется аналогичной кратности. К примеру, в *WiMAX* 802.16-2004 используется *FFT*256 с 200 активными поднесущими. [17, 19, 20]

Современные стандарты цифровой связи, такие как *WiMAX* и *LTE* используют технологию *OFDM* т.к. она позволяет решить проблемы коррекции АЧХ и ФЧХ канала связи, а также устойчива к многолучевому распространению радиосигнала. [21, 22]

Режим *OFDM* – это метод модуляции потока данных в одном частотном канале. При модуляции данных посредством ортогональных поднесущих в частотном канале выделяются N поднесущих так, что $f_k = f_c + k\Delta f$, где k – целое число из диапазона $[-N/2, N/2]$ (в данном случае $k \neq 0$). Расстояние между ортогональными поднесущими $\Delta f = 1/T_b$, где T_b – длительность передачи данных в символе. [23]

Помимо данных *OFDM*-символ включает защитный интервал длительностью T_g , так что общая длительность *OFDM*-символа $T_s = T_b + T_g$ (см. рисунок 1.1.3). Защитный интервал представляет собой копию оконечного фрагмента символа. Его длительность T_g может составлять 1/4, 1/8, 1/16 и 1/32 от T_b .

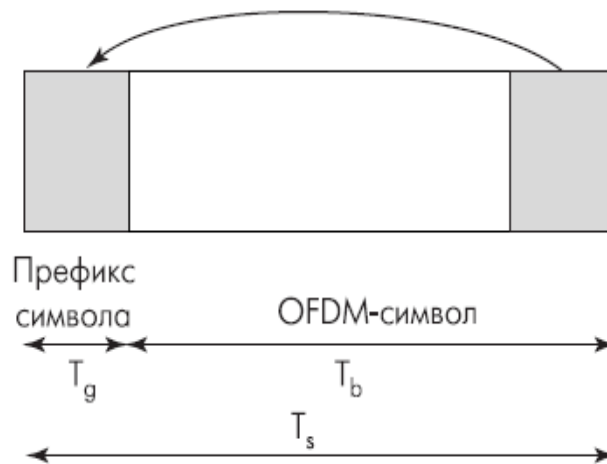


Рисунок 1.1.3 - *OFDM* – символ

Каждая поднесущая модулируется независимо посредством квадратурной амплитудной модуляции. Общий сигнал вычисляется методом обратного быстрого преобразования Фурье (ОБПФ) как

$$s(t) = \text{Re} \left\{ e^{i2\pi f_c t} \sum_{k=-N/2}^{N/2} c_k e^{i2\pi k\Delta f(t-T_g)} \right\}$$

($0 < t < T_s$), где c_k – комплексное представление символа квадратурной модуляции (*QAM*-символа). Комплексное представление удобно, поскольку генерация радиосигнала происходит с помощью квадратурного модулятора в соответствии с выражением $s_k(t) = I_k \cos(2\pi f_c t) - Q_k \sin(2\pi f_c t)$, где I_k и Q_k –

синфазное и квадратурное (целое и мнимое) значения комплексного символа, соответственно. [17]

Для работы алгоритмов БПФ/ОБПФ оптимально, чтобы количество точек соответствовало 2^m . Поэтому число поднесущих выбирают равным минимальному числу $N_{FFT} = 2^m$, превосходящему N . В режиме *OFDM* стандарта *IEEE 802.16* $N = 200$, соответственно $N_{FFT} = 256$. Из них 55 ($k = -128 \dots -101$ и $101 \dots 127$) образуют защитный интервал на границах частотного диапазона канала. Центральная частота канала ($k=0$) и частоты защитных интервалов не используются (т.е. амплитуды соответствующих им сигналов равны нулю).

Из остальных 200 поднесущих восемь частот – пилотные (с индексами $\pm 88, \pm 63, \pm 38, \pm 13$), остальные разбиты на 16 подканалов по 12 поднесущих в каждом, причем в одном подканале частоты расположены не подряд. Например, подканал 1 составляют поднесущие с индексами -100, -99, -98, -37, -36, -35, 1, 2, 3, 64, 65, 66. Деление на подканалы необходимо, поскольку в режиме *WirelessMAN-OFDM* предусмотрена (опционально) возможность работы не во всех 16, а в одном, двух, четырех и восьми подканалах – некий прообраз схемы множественного доступа *OFDMA*. Для этого каждый подканал и каждая группа подканалов имеют свой индекс (от 0 до 31).

Длительность полезной части T_b *OFDM*-символа зависит от ширины полосы канала BW и системной тактовой частоты (частоты дискретизации) F_s ; $F_s = N_{FFT}/T_b$. Соотношение $F_s/BW = n$ нормируется, и в зависимости от ширины полосы канала принимает значения 86/75 (BW кратно 1,5 МГц), 144/125 (BW кратно 1,25 МГц), 316/275 (BW кратно 2,75 МГц), 57/50 (BW кратно 2 МГц) и 8/7 (BW кратно 1,75 МГц и во всех остальных случаях).

Защитный интервал при *OFDM*-модуляции – мощное средство борьбы с межсимвольными помехами (межсимвольной интерференцией, МСИ), возникающими вследствие неизбежных в городских условиях переотражений и многолучевого распространения сигнала. МСИ приводит к тому, что в приемнике на прямо распространяющийся сигнал накладывается

переотраженный сигнал, содержащий предыдущий символ. При модуляции *OFDM* переотраженный сигнал попадает в защитный интервал и вреда не причиняет. Однако этот механизм не предотвращает внутрисимвольную интерференцию – наложение сигналов с одним и тем же символом, пришедших с фазовой задержкой. В результате информация может полностью исказиться или (например, при фазовом сдвиге на 180°) просто исчезнуть. Для предотвращения потери информации при пропадании отдельных символов или их фрагментов стандарт *IEEE 802.16* предусматривает эффективные средства канального кодирования.

Кодирование данных на физическом уровне включает три стадии – рандомизацию, помехозащищенное кодирование и перемежение. Рандомизация – это умножение блока данных на псевдослучайную последовательность (ПСП), которую формирует генератор ПСП с задающим полиномом вида $1+x^{14}+x^{15}$.

Кодирование данных предполагает каскадный код с двумя стадиями – кодер Рида-Соломона из поля Галуа $GF(256)$ и сверточный кодер. [24] В базовом виде код Рида-Соломона оперирует блоками исходных данных по 239 байт, формируя из них кодированный блок размером 255 байт (добавляя 16 проверочных байт). Такой код способен восстановить до 8 поврежденных байт.

После кодирования следует процедура перемежения – перемешивания битов в пределах блока кодированных данных, соответствующего *OFDM*-символу. Эта операция проводится в две стадии. Цель первой – сделать так, чтобы смежные биты оказались разнесенными по несмежным поднесущим. На второй стадии смежные биты оказываются разнесенными в разные половины последовательности. Все это делается для того, чтобы при групповых ошибках в символе повреждались несмежные биты, которые легко восстановить при декодировании.

После перемежения начинается стадия модуляции. Исходя из выбранной схемы модуляции (*BPSK / QPSK / 16-QAM / 64-QAM*), блок

представляется в виде последовательности групп бит, соответствующих модуляционным символам (по 1 / 2 / 4 / 6 бит). Каждой группе ставится в соответствие значения Q и I из векторных диаграмм Грея (рисунке 1.1.4), которые затем используются при непосредственной модуляции поднесущей. Для усреднения амплитуд квадратурных символов используются нормализованные значения Q и I , т.е. умноженные на коэффициенты c (для QPSK $c = 1/\sqrt{2}$, для 16-QAM $c = 1/\sqrt{10}$, для 64-QAM $c = 1/\sqrt{42}$).

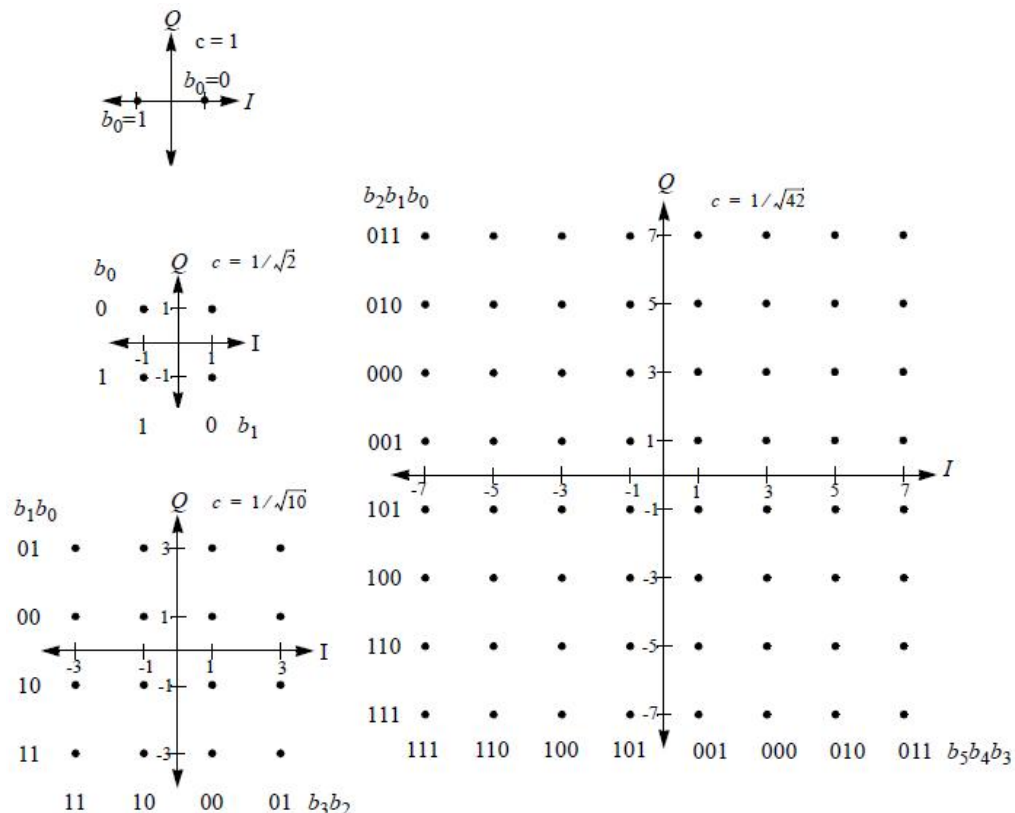


Рисунок 1.1.4 - Векторные диаграммы Грея (представление модуляционных символов) для BPSK, QPSK, 16-QAM и 64-QAM

Пилотные поднесущие модулируются посредством BPSK. Значения сигналов на этих поднесущих определяются на основании бинарной ПСП w_k с задающим полиномом $x^{11} + x^9 + 1$, причем в нисходящем субкадре k – номер символа относительно начала кадра, в восходящем – номер символа относительно начала пакета (рисунке 1.1.5). Инициализирующие слова генератора ПСП для нисходящего и восходящего потоков различны ($8FF_{16}$ и 555_{16} , соответственно). Собственно значения BPSK-символов вычисляются как $c_{-88}=c_{-38}=c_{63}=c_{88}=1-2w_k$; $c_{-63}=c_{-13}=c_{13}=c_{38}=1-2w_k$ в нисходящем канале и

$c_{-88}=c_{-38}=c_{13}=c_{38}=c_{63}=c_{88}=1-2w_k$; $c_{-63}=c_{-13}=1-2w_k$ – в восходящем.

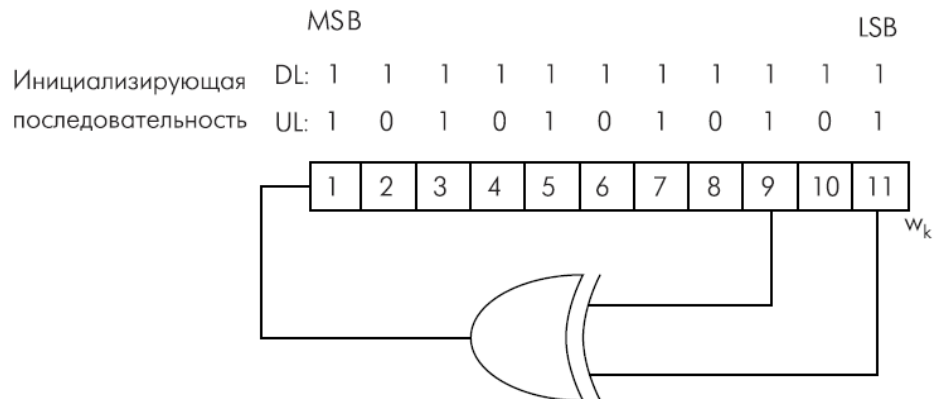


Рисунок 1.1.5 - Генерация модулирующей последовательности для пилотных поднесущих

После определения модуляционных символов посредством ОБПФ вычисляется сам радиосигнал и передается в передатчик. При приеме все процедуры производят в обратном порядке.

Для достижения высокой скорости передачи данных требуется выполнять процедуры ЦОС, такие как преобразование Фурье, канальное кодирование/декодирование и т.д. за время менее 1 *OFDM* символа. Например, для стандарта *WiMAX* 802.16 – 2004 - *OFDM* для полосы сигнала 7 МГц, частота дискретизации должна быть равна $256/200 \cdot 7 = 8,96$ МГц, т.е. при условии длины циклического префикса 32 длительность *OFDM* символа будет равна $288/8,96 = 32,1$ мкс. Такое быстодействие можно получить только в случае реализации алгоритмов ЦОС на аппаратном уровне в цифровой микросхеме: заказной или ПЛИС. Для этого необходимо реализовать алгоритмы на языке описания аппаратуры (*Verilog*, *VHDL*). [25, 26, 27, 28]

Для проводных каналов связи применяется очень похожая на технологию *OFDM* технология *DMT*.

Модуляция *DMT*

DMT (*Discrete Multi - Tone modulation* - дискретная многочастотная (многоканальная) модуляция) – метод передачи сигналов, в котором полная полоса пропускания делится между 255 поднесущими или подканалами с шириной полосы пропускания в 4 кГц каждая. Первый канал поднесущей

используется для передачи традиционного голосового сигнала и сети *POTS*. Данные *upstream* обычно передаются по каналам 7-32 (26-128 кГц), а данные *downstream* – по каналам 33-250 (138-1100 кГц). В действительности, метод *DMT* является разновидностью уплотнения *FDM*. Поток входящих данных делится на N каналов, имеющих одинаковую пропускную способность, но разную среднюю частоту несущей. Использование нескольких каналов с узкой полосой пропускания дает следующие преимущества:

- какими бы ни были характеристики линии, все каналы остаются независимыми, поэтому их можно декодировать по отдельности;
- при использовании *DMT* коэффициент передачи подбирается таким образом, чтобы каждый канал при наличии шума мог функционировать независимо; в этом методе изменяется количество битов на подканал или тон. В результате снижается общее воздействие шума при импульсной помехе на постоянной частоте.

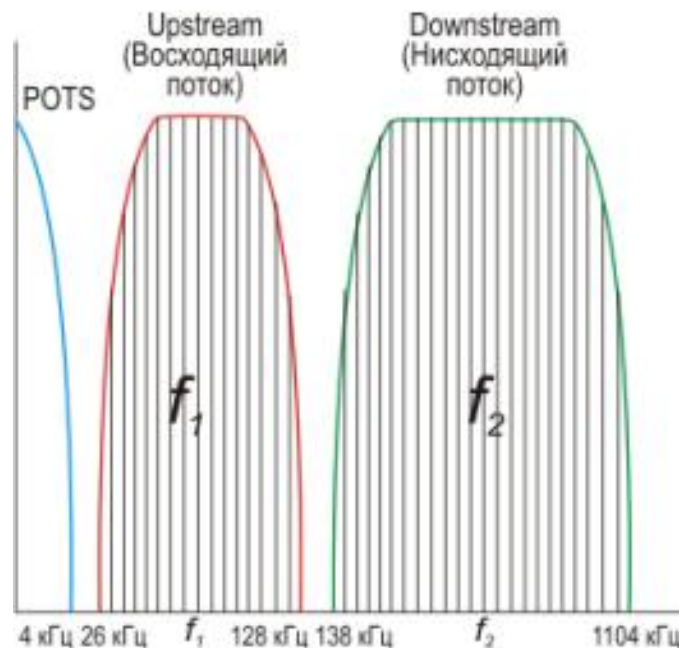


Рисунок 1.1.6 - Частотный спектр для модуляции *DMT*

Основными характеристиками метода *DMT* являются:

- в методе используется мультиплексирование *FDM*, тесно связанное с ортогональным частотным разделением с мультиплексированием (*Orthogonal Frequency Division Multiplexing* - *OFDM*), как и в *DVB-T/H*;

- метод оговорен в стандарте T1.413, разработанном Национальным институтом стандартизации США (*American National Standards Institute - ANSI*);
- в канале заданы 256 подканалов;
- полоса пропускания каждого подканала равна 4,3125 кГц;
- каждый подканал независимо моделируется с помощью метода дискретной модуляции *QAM*;
- коэффициент усиления (спектральная плотность) каждого подканала составляет 16 бит/с/Гц для теоретического значения пропускной способности, равного 64 кбит/с;
- теоретическая пропускная способность для данных с полосой пропускания 1,104 МГц равна 16,384 Мбит/с;
- в стандартах *ITU 992.1 (G.dmt)*, *ITU 992.2 (G.lite)* и *ANSI T 1.431 Issue 2* оговорено использование различных вариантов и реализаций каналов *ADSL*, основанных на методе кодирования *DMT*;
- метод *DMT* был принят комитетом *ANSI T1* как стандарт кодирования для линий связи и используется в системах передачи сигналов по каналам *ADSL*.

На рисунке 1.1.6 показан частотный спектр для модуляции *DMT*. [29]
 Отличие *DMT* от *OFDM* симметричный спектр относительно частоты дискретизации пополам. [30]

Для $i > N/2$ реальная и мнимая части спектра имеют следующие значения:

$$S_{Re}[N-i] = S_{re}[i];$$

$$S_{Im}[N-i] = - S_{im}[i].$$

Специалисты компании *Fujitsu* разработали первую в мире технологию оптической передачи, позволяющую получить скорость 100 Гбит/с с использованием широкодоступных компонентов, рассчитанных на скорость 10 Гбит/с. [31]

Ключом к успеху стало применение дискретной многочастотной или многотональной модуляции (*Discrete Multi-Tone, DMT*), реализуемой с помощью цифрового сигнального процессора. Это позволило получить скорость 100 Гбит/с в канале с использованием компонентов, обычно обеспечивающих скорость 10 Гбит/с в расчете на один канал. Таким образом, в случае использования оптического приемопередатчика с четырьмя каналами можно получить скорость 400 Гбит/с (см. рисунок 1.1.7). Повышение скорости необходимо для вычислительных центров следующего поколения, чтобы улучшить поддержку облачных сервисов. [32]

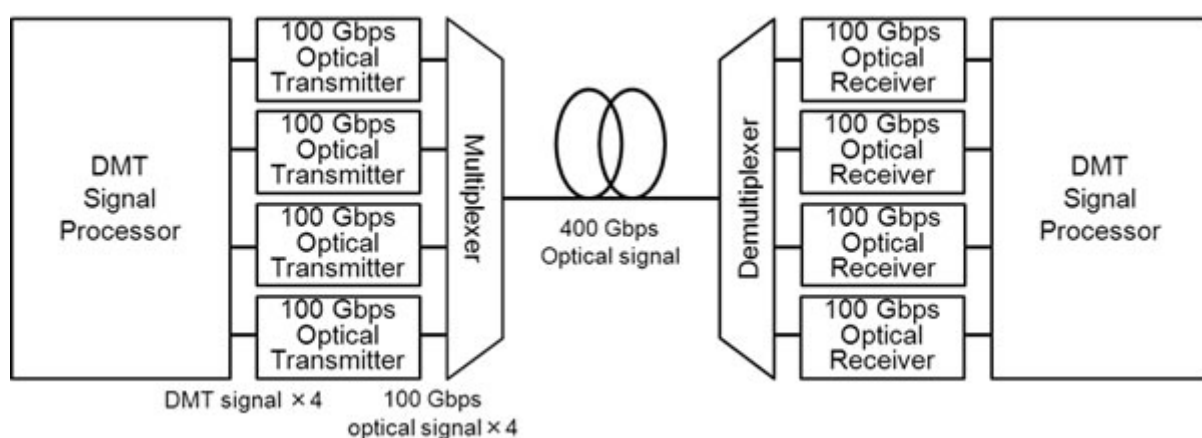


Рисунок 1.1.7 - Мультиплексирование 4-х потоков

Вышеупомянутая модуляция *DMT* используется в *xDSL*, но в оптических приемопередатчиках, работающих со скоростью 100 Гбит/с, она применена впервые. Принцип *DMT* построен на разделении данных по большому количеству поднесущих и многоуровневой модуляции каждой поднесущей. Для оптимизации параметров был разработан алгоритм, оценивающий и учитывающий перед началом передачи состояние линии связи.

В расчете на практическое применение разработки специалисты *Fujitsu* рассматривают возможность создания интегрированного модема *DMT*.

1.2. Синхронизация приёмо-передающих устройств в *OFDM* системе

Различие частоты дискретизации сигналов в передатчике и приемнике приводит к изменению масштаба сигнала по времени и ширине спектра. [33]

Функциональная схема системы передатчик-канал-приемник представлена на рисунке 1.2.1.

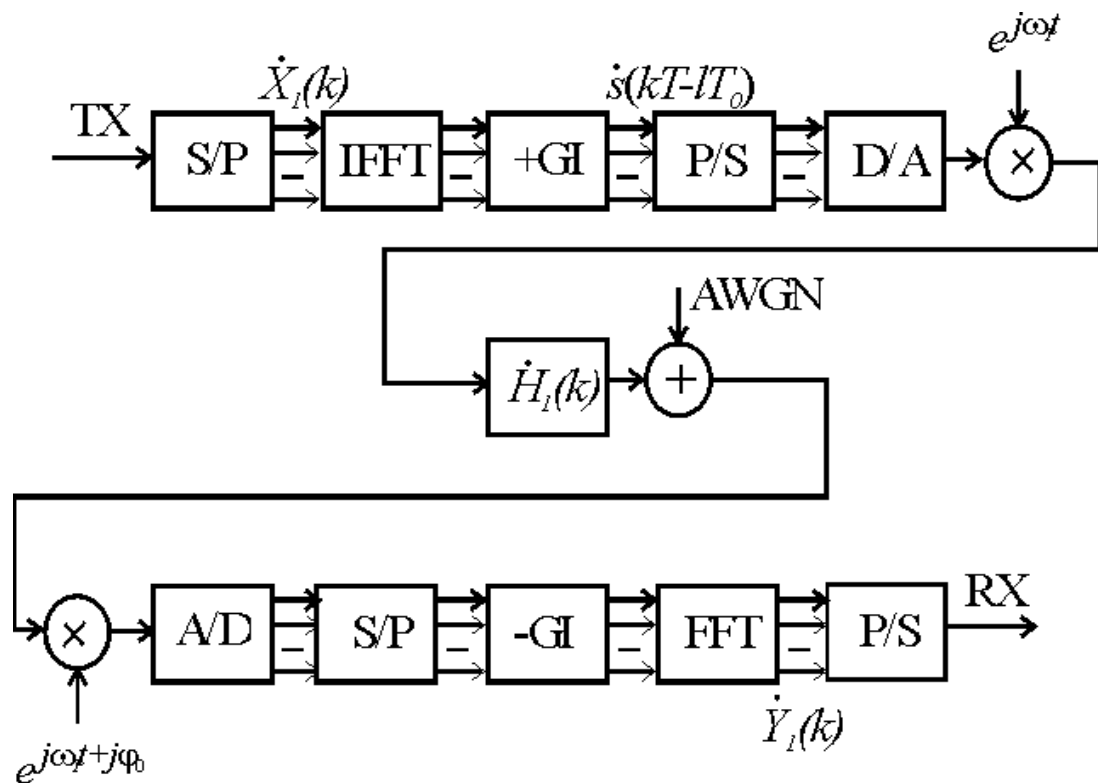


Рисунок 1.2.1 - Функциональная схема системы передатчик-канал-приемник, применяющей *OFDM*

На рисунке 1.2.1 обозначено: *S/P* – преобразователь последовательных данных в параллельные; *IFFT* – блок обратного быстрого преобразования Фурье; *+GI* – формирователь защитного интервала; *P/S* – преобразователь параллельных данных в последовательные, *D/A* – цифро-аналоговый преобразователь (ЦАП); *TX* и *RX* – передаваемые и принимаемые данные.

Шаги синхронизации известные в литературе

1. Грубая временная синхронизация (из сигнала во временной области).
2. Корректировка смещения по времени (схемотехнически).
3. Точная синхронизация по частоте (из сигнала во временной области).
4. Корректировка смещения по частоте (схемотехнически).
5. Грубая синхронизация по частоте (из сигнала в частотной области).
6. Корректировка смещения по частоте (схемотехнически).

7. Оставшаяся временная синхронизация (из сигнала в частотной области).
8. Корректировка смещения по времени (математически).
9. Оценка ЧХ канала передачи.
10. Компенсация влияния ЧХ канала (математически).

Этапы 1 и 2 реализуются путем обработки *OFDM*-сигнала во временной области.

Для реализации этапов 4 и 5 необходимо получение дискретных отсчетов поднесущих при помощи дискретного преобразования Фурье.

Один из наиболее эффективных алгоритмов для оценки смещения по частоте во временной области базируется на взаимных корреляционных свойствах защитного интервала и символа. Алгоритм не требует предварительного вычисления БПФ. Как показали исследования, его точность достаточна для большинства практических приложений. [34]

Другой метод синхронизации рассмотрен в [35]. В данном способе используется синхронизация в частотной области, которая обеспечивается за счет грубой частотной и точной временной синхронизации. Алгоритмы для оценки смещения по частоте базируется на распознавании постоянных пилот-поднесущих. Метод оценки смещения по времени основан на измерении фазового рассогласования между соседними переменными пилот-поднесущими.

Временная синхронизация

Под временной синхронизацией в сетях *WiMAX* понимается поиск и анализ преамбулы в кадре нисходящего канала (*Downlink*) от базовой станции к абонентской. Обработка осуществляется в основной полосе частот с возможным цифровым понижением частоты вниз.

Все алгоритмы временной синхронизации основаны на расчете решающей функции по различным участкам преамбулы во временной или частотной области. [36, 37] Для обнаружения преамбулы можно использовать следующие свойства преамбулы: наличие циклического префикса; симметричность относительно центрального отсчета; наличие трех

«приблизительно» одинаковых участков во временной области (псевдопериодичность). [38, 41]

В предложенном в данной работе методе временная синхронизация осуществляется за счет расчета взаимной корреляционной функции, через БПФ и ОБПФ – синхронизация в середину циклического префикса (см. подробнее п. 2.4.).

В предложенном в данной работе способе тактовая и частотная синхронизация осуществляется программно-аппаратно за счет изменения управляющего напряжения на ГУН и подстройки единой опорной частоты (см. подробнее п. 2.5.).

Обнаружение по циклическому префиксу

Расчет решающей функции по циклическому префиксу осуществляется по формуле (1.2.1):

$$r_{CP}[d] = \left| \sum_{k=0}^{CP_L-1} x[d+k]x^*[d+k+Nfft] \right|, \quad (1.2.1)$$

где $x[d]$ — d -й отсчет входного сигнала на частоте дискретизации 11,2 МГц, $Nfft$ — размерность БПФ (для *WiMAX/Yota* — 1024). Длина циклического префикса CP_L в системе *Yota* равна 128 отсчетам. [38]

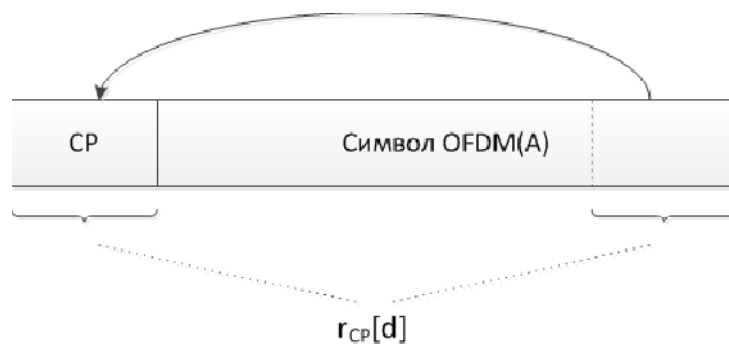


Рисунок 1.2.2 - Расчет решающей функции по циклическому префиксу

При расчете по формуле (1.2.1) анализируются два участка сигнала длиной 128 отсчетов, разнесенные во времени на 1024 отсчета (рисунок 1.2.2). На рисунке 1.2.3 представлена зависимость решающей функции по циклическому префиксу от порядкового номера отсчета при анализе последовательности *OFDMA*-символов.

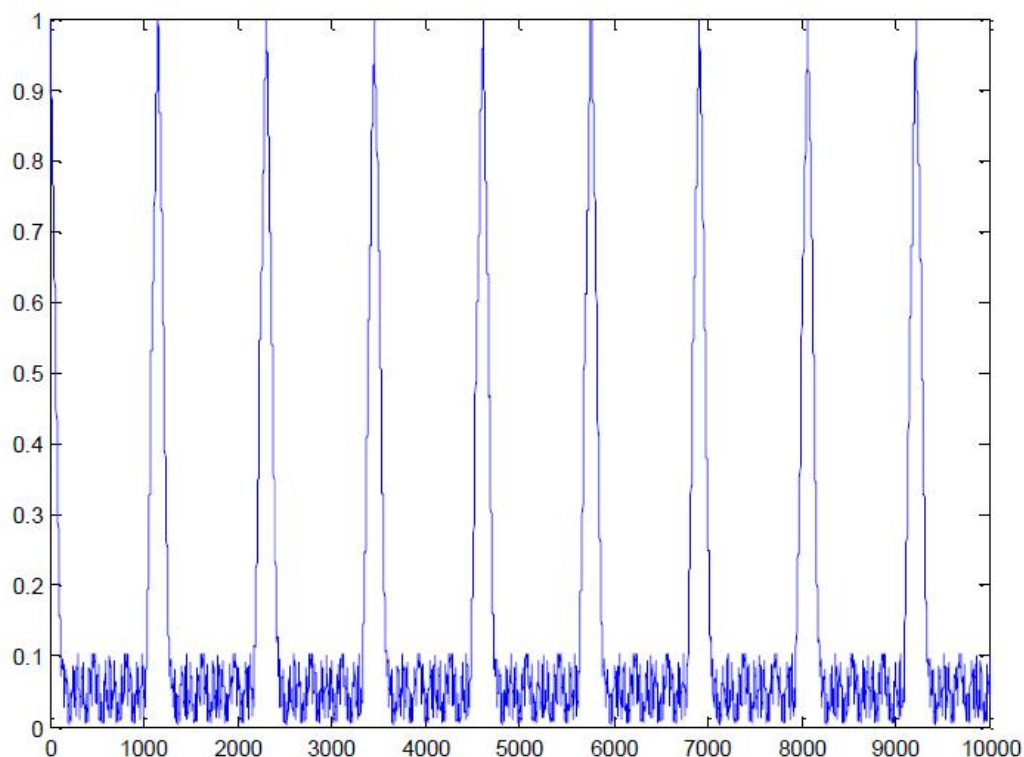


Рисунок 1.2.3 - Зависимость решающей функции по циклическому префиксу от порядкового номера отсчета при анализе последовательности *OFDMA*-СИМВОЛОВ

Обнаружение по половинам преамбулы (асимметричный алгоритм)

Расчет решающей функции между половинам преамбулы осуществляется по формуле (1.2.2):

$$r_{as}[d] = \left| \sum_{k=1}^{N_{fft}/2-1} x[d+k]x[d-kt] \right|, \quad (1.2.2)$$

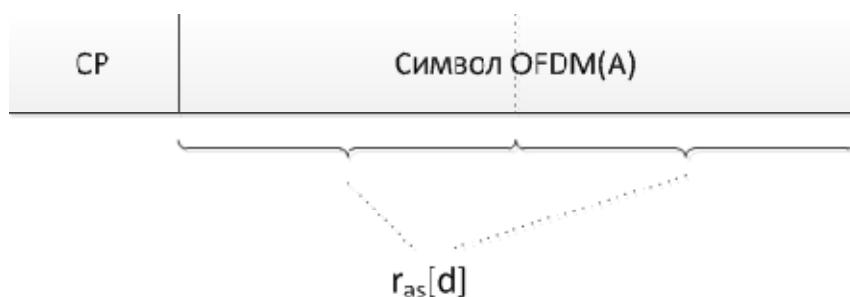


Рисунок 1.2.4 - Расчет решающей функции по половинам преамбулы

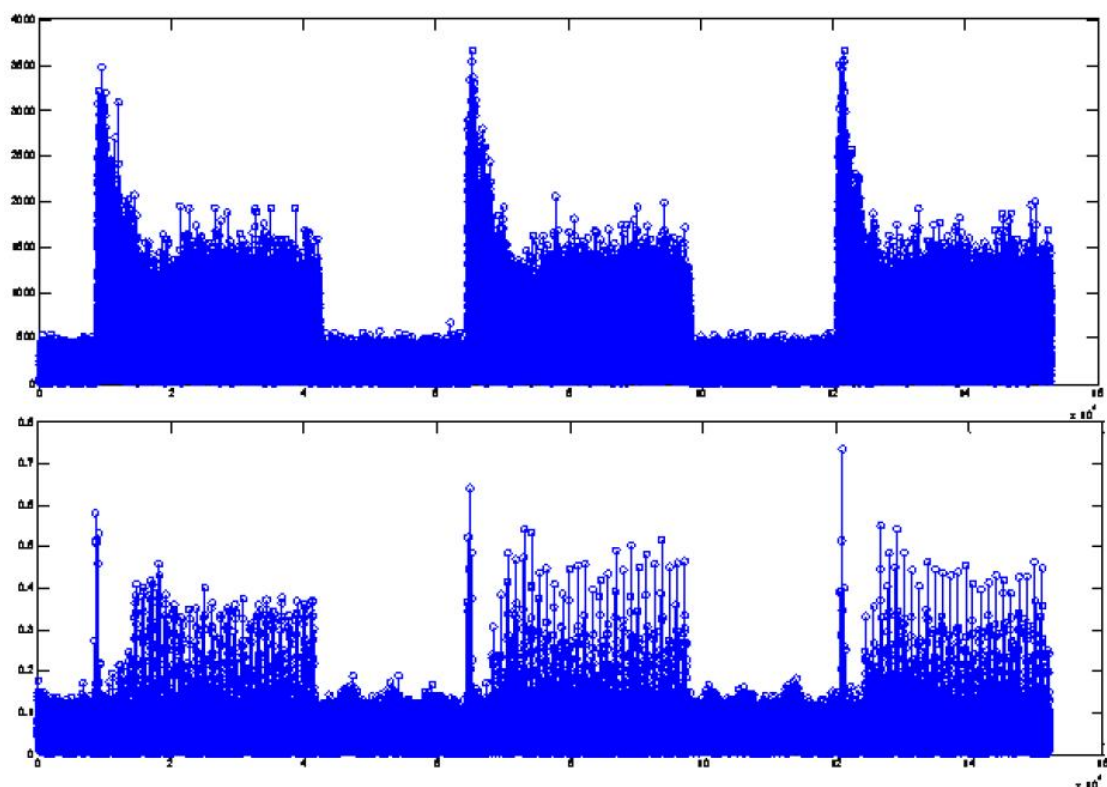


Рисунок 1.2.5 - Зависимость решающей функции по половинам преамбулы от порядкового номера отсчета при анализе записи реального сигнала

При расчете по формуле (1.2.2) анализируются два соседних участка сигнала длиной 511 отсчетов (рисунок 1.2.4). При этом используется свойство вещественности преамбулы в частотной области. Первый и 513-й отсчеты не учитываются при анализе, так как они не имеют пары для симметрии.

Возможна модификация данного алгоритма, в которой аналогичным образом анализируется циклический префикс преамбулы и ее первые 128 отсчетов.

На рисунке 1.2.5 представлены огибающая сигнала системы *WiMAX/Yota* и результаты расчета решающей функции по половинам преамбулы. Огибающая входного сигнала – вверху, решающая функция – внизу.[38]

Обнаружение по повторяющимся участкам (периодический алгоритм)

Расчет решающей функции по повторяющимся участкам преамбулы осуществляется по формуле (1.2.3):

$$r_T[d] = \left| \sum_{k=0}^{\text{floor}(Nfft/3)-1} x[d+k]x^*[d+k+\text{floor}(Nfft/3)] \right|, \quad (1.2.3)$$

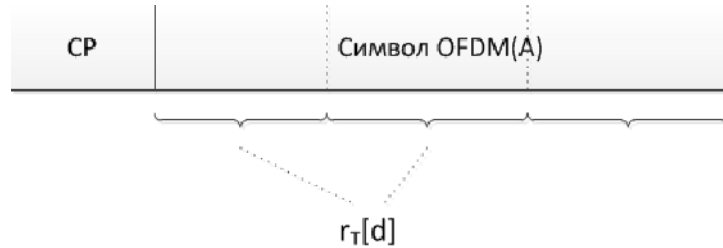


Рисунок 1.2.6 - Расчет решающей функции по повторяющимся участкам преамбулы

При расчете по формуле (1.2.3) анализируются два соседних участка сигнала длиной $[1024/3]$ отсчета (рисунок 1.2.6). При этом применяется свойство преамбулы, заключающееся в использовании в частотной области только каждой третьей поднесущей. Вследствие наличия трех «примерно» одинаковых участков в сигнале зависимость решающей функции от времени будет иметь области неопределенности в виде «площадок» (рисунок 1.2.7). [38]

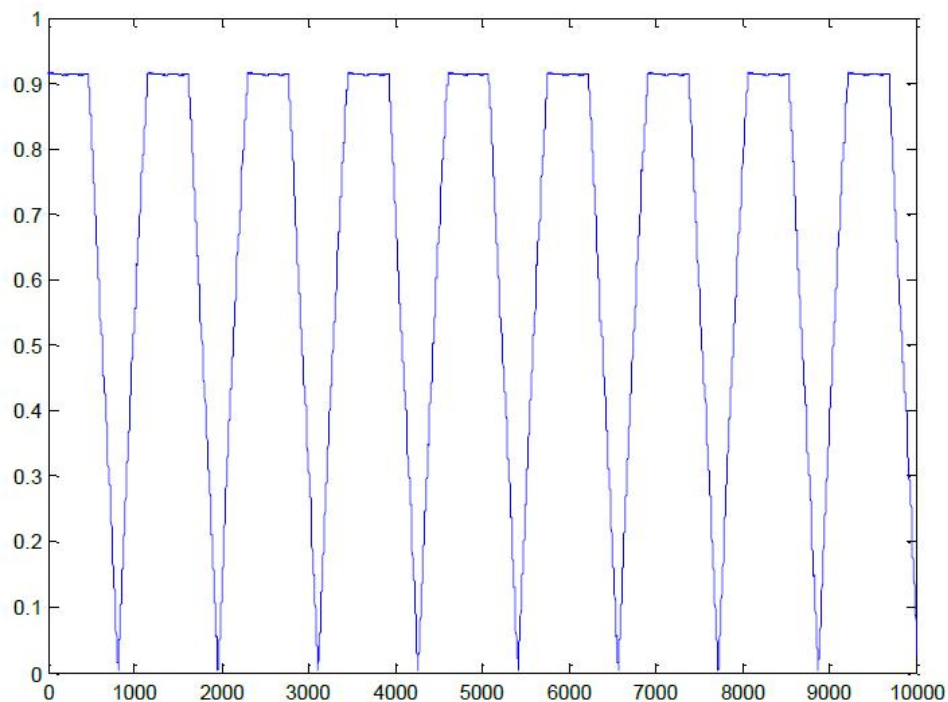


Рисунок 1.2.7 - Зависимость решающей функции по повторяющимся участкам преамбулы при анализе последовательности *OFDMA*-символов преамбул

Обнаружение в частотной области

При реализации процедуры обнаружения преамбулы в частотной области учитывается то, что в преамбуле используется только каждая третья поднесущая.

Расчет решающей функции в частотной области осуществляется по формуле:

$$R_f = \min_{i=0,1,2} \left\| \vec{r}_f - \frac{\left\| \vec{h}_i \right\|}{\left\| \vec{r}_f \right\|} \vec{h}_i \right\|^2, \quad i = 0, 1, 2, \quad (1.2.4)$$

где $\vec{r}_f = r_0, r_1, \dots, r_{N_{ff}-1}$ - вектор отсчетов сигнала, подлежащего проверке, в частотной области, $\vec{h}_i = h_{i0}, h_{i1}, \dots, h_{iN_{ff}-1}^T$, — вектор мощностей поднесущих в идеальной преамбуле, $h_{ij} = 0, 1$ — квадрат амплитуды j -й поднесущей в i -м сегменте, $\left\| \vec{x} \right\|$, — евклидова норма вектора $\vec{x}$.

С помощью дроби в выражении (1.2.4) осуществляется нормировка по мощности входного сигнала в частотной области.

Данный алгоритм обладает самой высокой сложностью по сравнению с другими алгоритмами обнаружения преамбулы при сходных вероятностных характеристиках.

Свойства методов обнаружения преамбулы

1. По циклическому префиксу: устойчивость к сдвигам по частоте, высокая помехоустойчивость, обнаружение всех символов *OFDMA* (как преамбулы, так и информационных), большая погрешность.

2. Асимметричный: устойчивость к сдвигам по частоте, средняя помехоустойчивость, высокая вероятность ложного обнаружения, низкая погрешность, чувствительность к ошибкам по частоте дискретизации.

3. Асимметричный по циклическому префиксу: то же, что и асимметричный, хуже помехоустойчивость, меньше сложность.

4. Периодический: высокая помехоустойчивость, устойчивость к сдвигам по частоте, очень большая погрешность.

5. По частотной области: очень большая сложность, средняя помехоустойчивость. [38]

Рассмотрим способы временной синхронизации известные в зарубежной литературе. Способ временной синхронизации, описанный в патенте США [39] №5,732,113 от 24.03.1998 авторов: *Schmidl T.M., Cox D.C., Timing and Frequency Synchronization of OFDM signals*. В упомянутом патенте предложен способ (метод Шмидл и Кокс) синхронизации параметров сигнала в *OFDM* системах, который основан на приеме преамбулы. Первый символ преамбулы, по которому осуществляется временная синхронизация, включает каждую 2-ую спектральную компоненту, т.е. данный символ во временной области состоит из двух идентичных частей и циклического префикса. В процессе синхронизации временное положение сигнала определяют по положению максимума решающей функции:

$$M(d) = |P(d)|^2 / (R(d))^2, \quad (1.2.5)$$

$$\text{где } P(d) = \sum_{i=0}^{\frac{N}{2}-1} \left(r^*(d+i) \cdot r\left(d+i+\frac{N}{2}\right) \right);$$

$$R(d) = \sum_{i=0}^{\frac{N}{2}-1} \left| r\left(d+i+\frac{N}{2}\right) \right|^2;$$

N – количество точек в преобразовании Фурье;

d – временной индекс соответствующий первому дискрету в окне из N дискрет;

r – комплексное значение оцифрованного принятого сигнала.

Главным недостатком метода является плоская вершина решающей функции (1.2.5), которая при наличии шумов приводит к значительным

ошибкам временной синхронизации. Временная метрика решающей функции без шумов и искажений для метода Шмидла и Кокса представлена на рисунке 1.2.8 (сплошная линия).

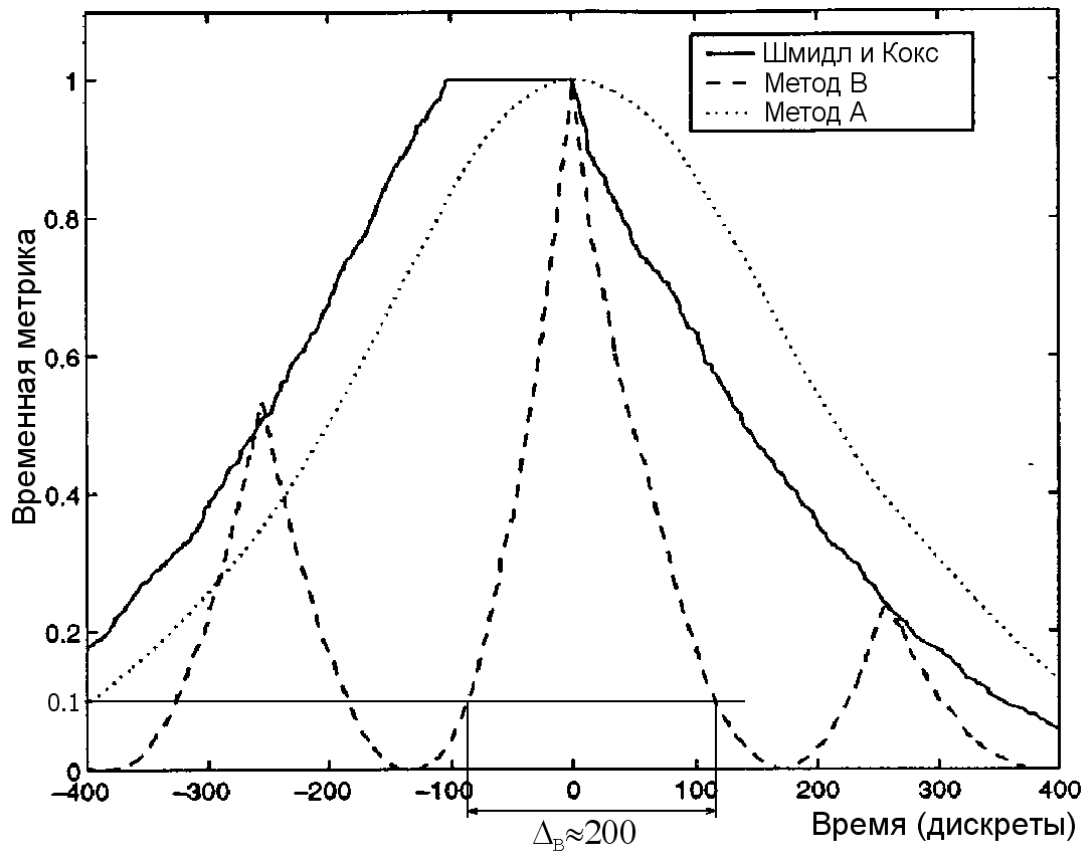


Рисунок 1.2.8 - Временная метрика без шумов и искажений

Также известен способ временной синхронизации, описанный в статье [40]. В указанной статье предложено два различных усовершенствования метода Шмидла и Кокса: метод А и метод В. Метод А также, как метод Шмидла и Кокса, основан на обработке символа преамбулы состоящей во временной области из двух идентичных частей и циклического префикса.

В методе А дополнительно используется свойство обязательного наличия циклического префикса длительностью N_g в *OFDM* символе. Оценку временной задержки входного сигнала определяют по положению максимума решающей функции (1.2.6). Временная метрика без шумов и искажений для метода А представлена на рисунке 1.2.8 пунктирной линией.

Решающая функция имеет вид:

$$M_1(d) = \frac{1}{N_g + 1} \sum_{k=-N_g}^0 M_f(d+k), \quad (1.2.6)$$

где N_g - длительность циклического префикса в дискретах;

$$M_f(d) = \frac{|P(d)|^2}{R_f^2(d)};$$

$$P(d) = \sum_{i=0}^{\frac{N}{2}-1} \left(r^*(d+i) \cdot r\left(d+i+\frac{N}{2}\right) \right);$$

$$R_f(d) = \frac{1}{2} \sum_{i=0}^{N-1} |r(d+i)|^2; \quad (1.2.7)$$

N – количество точек в преобразовании Фурье;

d – временной индекс, соответствующий первому дискрету в окне из N дискрет;

r – комплексное значение оцифрованного принятого сигнала.

Временная синхронизация системы связи методом В заключается в следующем:

на передающей стороне формируют преамбулу из *OFDM* символов, модулированных по схеме квадратурной фазовой манипуляции, содержащих циклический префикс, первый символ которой используют для определения по нему временного положения сигнала; сигнал преамбулы переносят на несущую частоту; усиливают и передают его по каналу связи;

на приемной стороне входной сигнал фильтруют, усиливают, переносят на нулевую промежуточную частоту, осуществляют его аналого-цифровое преобразование, формируя входной цифровой комплексный сигнал на нулевой промежуточной частоте;

далее формируют решающую функцию (см. временную метрику без шумов и искажений для метода В на рисунке 1.2.8 – штриховая линия), определяемую формулой (1.2.8);

определяют максимум M решающей функции $M_2(d)$ и по его положению оценивают временную задержку сигнала;

первый символ преамбулы, используемый для определения по нему временного положения сигнала, состоит из четырех одинаковых по длине частей, первые две из которых идентичны, а последние две негативны по отношению к первым двум.

$$M_2(d) = |P_2(d)|^2 / R_2^2(d), \quad (1.2.8)$$

$$\text{где } P_2(d) = \sum_{k=0}^1 \sum_{m=0}^{L-1} r^*(d + 2Lk + m) \cdot r(d + 2Lk + m + L);$$

$$R_2(d) = \sum_{k=0}^1 \sum_{m=0}^{L-1} |r(d + 2Lk + m + L)|^2;$$

$$L = N/4;$$

N – количество точек в преобразовании Фурье;

d – временной индекс соответствующий первому дискрету в окне из N дискрет;

r – комплексное значение оцифрованного принятого сигнала.

На рисунке 1.2.9 представлена зависимость дисперсии оценки временного положения сигнала от отношения сигнал/шум для метода Шмидла и Кокса, метода Шмидла и Кокса с использованием формулы (1.2.7), метода А, метода В, и метода В с использованием формулы (1.2.7). Наименьшее значение дисперсии наблюдается у метода В.

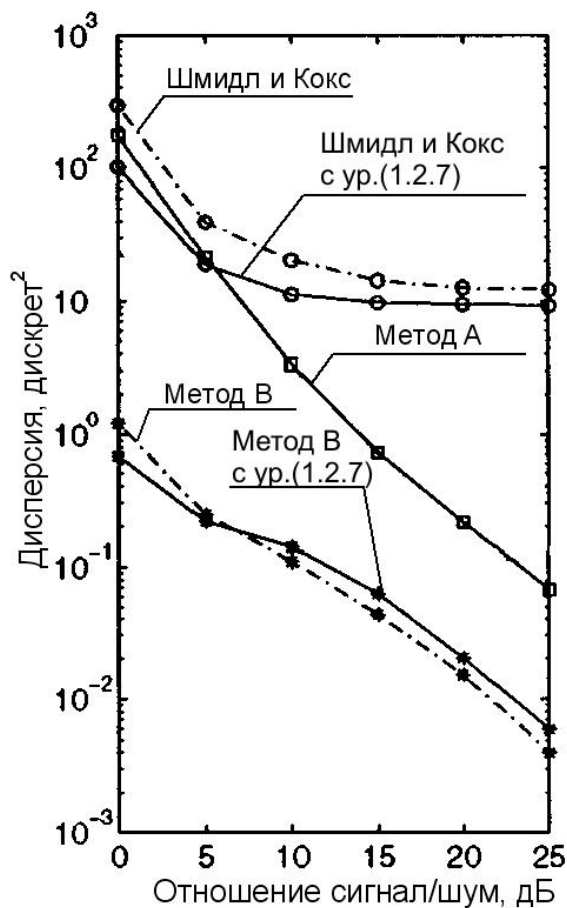


Рисунок 1.2.9 - Зависимость дисперсии оценки временного положения сигнала от отношения сигнал/шум

Недостатком рассмотренных методов является низкая точность при значительных шумах, которая обусловлена применением решающей функции с плоским и относительно плоским максимумом.

1.3. Обзор методов коррекции комплексной передаточной характеристики

Коррекция передаточной характеристики канала связи обычно осуществляется в два шага. На первом шаге осуществляется оценка коэффициента передачи канала в данной частотной позиции в данный момент времени тем или иным способом. На втором шаге осуществляется интерполяция полученных оценок передаточной характеристики канала во времени и по частоте.

Для оценки коэффициентов передачи применяются метод наименьших квадратов, минимума среднего квадрата ошибки, адаптивные алгоритмы, алгоритмы, на основе фильтра Калмана и др. Различают методы одномерные (для оценки используют информацию только по одной из осей, времени или частоты) и двумерные (используют информацию по обеим осям).

Интерполяция полученных оценок передаточной характеристики во времени и/или частоте может быть линейной, кубической, сплайн и т. п. Более того, в более общем случае для восстановления передаточной характеристики допускается применение различных методов аппроксимации, а не интерполяции.

Эквалайзер по преамбуле

Оценка частотной характеристики канала производится по пилотным отсчетам преамбулы с последующей линейной интерполяцией. Применение линейной интерполяции позволяет реализовать алгоритм относительно просто (по сравнению с применением нелинейной интерполяции). Полученная оценка характеристики канала применяется для коррекции всех символов обрабатываемого кадра, после чего происходит ее пересчет по следующей преамбуле.

Данный алгоритм не предназначен для работы в условиях быстрых замираний, так как при этом реальная частотная характеристика канала значительно изменяется от символа к символу, что приводит к возрастающей ошибке оценки передаточной характеристики канала. В то же время, из-за относительно большого количества пилотных отсчетов в преамбуле данный алгоритм позволяет эффективно устранять фазовый набег в частотных каналах при наличии ошибок тактовой синхронизации. Заметим, что эквалайзер по преамбуле может быть использован в составе других алгоритмов для вычисления оценочного значения величины ошибки тактовой синхронизации.

Алгоритм эквалайзера по преамбуле требует наименьшего количества аппаратных ресурсов среди всех рассматриваемых алгоритмов, так как

относительно простое вычисление оценки передаточной характеристики производится только один раз за все время длительности кадра. [38]

Эквалайзер по ближайшим пилотным поднесущим одного *OFDMA*-символа

В таком эквалайзере оценка частотной характеристики канала производится по пилотным отсчетам *OFDMA*-символов, при этом интерполяция не производится. Оценочное значение передаточной характеристики, вычисленное по каждой пилотной поднесущей *OFDMA*-символа, считается одинаковым для всех ближайших к нему информационных поднесущих этого символа. Перерасчет характеристики происходит для каждого символа.

Данный алгоритм является самым простым из рассматриваемых для реализации вследствие отсутствия интерполяции, однако он показывает одни из худших результатов по уровню средней вероятности ошибки на бит и устойчивости к ошибкам тактовой синхронизации. [38]

Эквалайзер по одному *OFDMA*-символу

Оценка частотной характеристики канала для каждого *OFDMA*-символа производится по пилотным поднесущим этого символа.

Интерполяция в этом случае может быть линейной, кубическими полиномами Эрмита или интерполяции кубическими сплайнами.

Наилучшие результаты показывает алгоритм, использующий интерполяцию сплайнами, наихудшие — алгоритм, использующий линейную интерполяцию. Максимальная разность в величине вероятности ошибки между наихудшим и наилучшим алгоритмами составляет порядка $\Delta P_{ош} = 8 \cdot 10^{-3}$. При использовании нелинейной интерполяции, по сравнению с линейной, резко возрастает сложность реализации алгоритма и его требования к аппаратным ресурсам. Тем не менее, необходимо отметить, что эквалайзеры, основанные на нелинейной интерполяции, показывают плохую устойчивость к ошибкам тактовой синхронизации, что приводит к необходимости усложнения последней. [38]

Эквалайзер по двум *OFDMA*-символам

Оценка передаточной характеристики канала для каждого символа производится по пилотным отсчетам двух символов: четного и нечетного. При этом используется свойство расположения пилотных поднесущих в кластерах сигналов *WiMAX* в зоне *PUSC* нисходящего канала. В каждом четном символе пилотные поднесущие располагаются на 1 и 13 позициях кластера, в каждом нечетном — на 5 и 9 позициях. При совмещении пилот-поднесущих двух последовательных символов с сохранением их позиций, формируется вектор, содержащий в два раза больше пилотных отсчетов по сравнению с вектором в алгоритме эквалайзера по одному символу. На следующем этапе обработки осуществляется линейная интерполяция и вычисление оценки частотной характеристики канала. Применение линейной интерполяции позволяет реализовать алгоритм относительно просто (по сравнению с применением нелинейной интерполяции).

Алгоритм эквалайзера по двум символам показывает одни из лучших результатов среди всех рассматриваемых алгоритмов как в канале с замираниями для скорости 3 км/ч, так и для 60 км/ч. Данный результат объясняется тем, что в результате совмещения пилот-поднесущих двух соседних символов количество информации о частотно-селективном характере замираний возрастает по сравнению со случаем эквалайзера по одному символу. При этом даже при скорости перемещения АС в 60 км/ч состояние канала во времени не успевает значительно измениться на протяжении двух символов, что обеспечивает малые погрешности оценки характеристики канала в результате совмещения пилот-поднесущих этих двух символов. [38]

Двумерный последовательный эквалайзер

Оценка частотной характеристики канала для каждого N -го символа производится с помощью двумерной линейной интерполяции по всем символам, начиная с нулевого и заканчивая N -м. Таким образом, происходит интерполяция как в частотной области, так и во временной, что теоретически

должно позволить получить дополнительную информацию о состоянии канала. Применение линейной интерполяции позволяет реализовать алгоритм относительно просто (по сравнению с применением нелинейной интерполяции).

Алгоритм двумерного последовательного эквалайзера обладает наибольшей сложностью реализации среди всех рассмотренных. [38]

Двумерный эквалайзер по целому кадру

В данном алгоритме оценка частотной характеристики канала производится после окончания приема целого кадра. При использовании вектора комплексных амплитуд пилотных поднесущих от каждого информационного символа формируется матрица комплексных амплитуд всего кадра, по которой производится двумерная линейная интерполяция. Результатом данной операции является набор оценок частотной характеристики канала для каждого символа кадра. Применение линейной интерполяции позволяет реализовать алгоритм относительно просто (по сравнению с применением нелинейной интерполяции).

Вследствие осуществления интерполяции как в частотной области, так и во временной, теоретически возможно получить дополнительную (по сравнению с алгоритмами с одномерной интерполяцией) информацию о состоянии канала. Однако результаты имитационного моделирования показывают, что вследствие замираний даже при скорости движения АС 3 км/ч частотная характеристика канала изменяется существенным образом на протяжении нескольких символов. Двумерная интерполяция в символах кадров *WiMAX PUSC* достоверно происходит только между всеми четными символами и, параллельно, между всеми нечетными символами, так как они имеют одинаковое расположение пилотных поднесущих. Результаты двумерной интерполяции по соседним символам оказываются менее надежными, так как она происходит между несовпадающими по позициям пилот-поднесущими. Также при быстром изменении во времени характеристики канала, двумерная интерполяция начинает вносить

дополнительные ошибки, сглаживая резкие изменения этой характеристики во времени.

В результате алгоритм двумерного эквалайзера по кадру показывает результаты примерно такие же, как алгоритм эквалайзера по ближайшим пилот-поднесущим. [38]

Двумерный эквалайзер по четырем *OFDMA*-символам

Принцип оценки частотной характеристики канала в данном случае сходен с принципом, применяемым в алгоритме двумерного последовательного эквалайзера. Однако линейная двумерная интерполяция для N -го символа происходит или на интервале $[N - 3, N]$ для $N \geq 4$, или на интервале $[0; N]$ для $N < 4$. Таким образом, интерполяция выполняется только по четырем соседним символам, что сделано для упрощения реализации алгоритма. Кроме того, учет большего числа символов оказывается бесполезен из-за влияния замираний в канале. Применение линейной интерполяции позволяет реализовать алгоритм относительно просто (по сравнению с применением нелинейной интерполяции).

Вследствие быстрого изменения во времени характеристики канала двумерная интерполяция начинает вносить дополнительные погрешности в оценку этой характеристики, сглаживая резкие изменения во времени.

Алгоритм двумерного эквалайзера по четырем символам показывает примерно такие же результаты, как алгоритм двумерного эквалайзера по кадру и алгоритм эквалайзера по ближайшим пилот-поднесущим. [38]

Эквалайзер по трем *OFDMA*-символам

Принцип оценки передаточной характеристики канала похож на принцип, применяющийся в алгоритме эквалайзера по двум символам. Используется свойство расположения пилотных поднесущих в кластерах сигналов *WiMAX* в зоне *PUSC*. В каждом четном символе пилотные поднесущие располагаются на 1 и 13 позициях кластера, в каждом нечетном — на 5 и 9 позициях. При совмещении пилот-поднесущих двух последовательных символов с сохранением их позиций формируется вектор,

содержащий в два раза больше отсчетов комплексных амплитуд пилотных поднесущих по сравнению с вектором в алгоритме эквалайзера по одному *OFDMA*-символу. Однако в данном случае для каждого N -го символа для построения вектора оценок передаточной характеристики канала используются как оценки, полученные по пилотным поднесущим данного символа, так и среднее арифметическое между оценками $(N - 1)$ -го и $(N + 1)$ -го символов. Среднее арифметическое вычисляется по вещественной и мнимой частям комплексных амплитуд пилотных поднесущих на совпадающих позициях в $(N-1)$ -м и $(N+ 1)$ -м символах. На следующем этапе обработки осуществляется линейная интерполяция и вычисление оценки передаточной характеристики канала. Применение линейной интерполяции позволяет реализовать алгоритм относительно просто (по сравнению с применением нелинейной интерполяции).

В данном случае интерполяция производится по вещественной и мнимой частям комплексных амплитуд пилотных поднесущих, что упрощает реализацию и повышает эффективность алгоритма, вследствие отсутствия неоднозначности при определении фазы, равной $2\pi n$, $n = 1, 2, \dots$. Таким образом, применяется аддитивный метод получения оценки характеристики канала.

Рассматриваемый алгоритм показывает лучшие результаты по сравнению с алгоритмом эквалайзера по двум символам и один из лучших результатов среди всех рассматриваемых алгоритмов. [38]

Комбинированный эквалайзер

Данный алгоритм является сочетанием двух алгоритмов — модифицированного алгоритма эквалайзера по преамбуле и алгоритма эквалайзера по трем символам. На первом этапе с помощью пилотных поднесущих преамбулы происходит оценка величины ошибки тактовой синхронизации (количество отсчетов). Для каждого i -го отсчета в частотной области вследствие ошибки при синхронизации на M отсчетов, существует фазовый набег, равный

$\Delta j_i = \frac{2p i M}{Nfft}$, где $Nfft$ — размерность преобразования Фурье.

Таким образом, между величиной фазового набег и позицией отсчета существует линейная зависимость вида $\Delta \varphi_i = k i$, где $k = 2\pi M/Nfft$ — тангенс угла наклона характеристики. Зная оценочное значение k , можно отыскать оценку M :

$$\bar{M} = \frac{\bar{k} Nfft}{2p}, \text{ где } \bar{k} \text{ и } \bar{M} \text{ — оценки. Вычисление } \bar{k} \text{ производится с}$$

помощью усреднения по всем отношениям приращения функции φ_i к приращению аргумента в точках, где эти значения точно известны (в точках расположения пилот-поднесущих преамбулы):

$$\left(\frac{\Delta j_{i+3} - \Delta j_i}{(i+3) - i} \right)$$

Таким образом:

$$\bar{M} = \frac{Nfft}{2p (N_{pp} - 1)} \sum_i \frac{\Delta j_{i+3} - \Delta j_i}{(i+3) - i},$$

где N_{pp} — количество комплексных амплитуд пилотных поднесущих в обрабатываемой преамбуле. После получения значения $\bar{M}$ производится построение вектора коррекции фазового набег. Каждый i -й его элемент определяется соотношением

$$corr(i) = \exp \left(\frac{j 2p (i + SI) \bar{M}}{Nfft} \right),$$

где $j = \sqrt{-1}$, SI — величина верхнего защитного интервала в отсчетах.

При последующей обработке каждого N -го символа вектор $corr$ применяется для предварительной коррекции фазовой характеристики символов с номерами $N - 1$, N и $N + 1$. Данный процесс может быть оптимизирован путем сохранения в регистрах обработанных N -го и $(N + 1)$ -го символов для их использования во время работы эквалайзера для $(N + 1)$ -го символа. После этого происходит обработка в соответствии с алгоритмом эквалайзера по трем символам.

В отсутствие ошибок синхронизации характеристика комбинированного алгоритма совпадает или немного лучше характеристики алгоритма эквалайзера по трем символам. Небольшой выигрыш достигается за счет устранения с помощью пилот-поднесущих преамбулы начального набега фазы. При возрастании величины ошибки синхронизации характеристика комбинированного алгоритма не изменяется в отличие от характеристик остальных рассмотренных алгоритмов, демонстрирующих тенденцию к повышению средней вероятности ошибки на бит при том же значении отношения сигнал/шум. [38]

Также известны другие методики коррекции АЧХ и ФЧХ. [42, 43, 44, 45]

1.4. Применение пилот-поднесущих для коррекции КПХ системы связи

В эквалайзере также применяются пилотные поднесущие, на их основе осуществляется фазовая синхронизация и коррекция комплексной передаточной характеристики системы связи. Пилот-поднесущие располагаются на специально выделенных для них частотах, которые равномерно распределены в полосе занимаемых частот. Информация, которая передается на пилотных поднесущих, заранее известна. На основе информации, полученной после анализа пилотных поднесущих переданных и принятых символов, можно определить комплексный коэффициент передачи и сдвиг фазы на данных частотах, а затем решить задачу аппроксимации комплексной передаточной характеристики системы связи во всей полосе занимаемых частот. [46]

Частоты пилот-поднесущих модулируются методом *BPSK*, их значения рассчитываются по формуле $c_k = 1 - 2w_k$ ($c_k = \pm 1$, определяется в зависимости от номера пилот-поднесущей по соответствующему стандарту, см. рисунок 1.4.1).

Генератор псевдослучайной двоичной последовательности, изображенный на рисунке 1.4.2, будет использоваться для получения последовательности w_k .

Полином для генератора псевдослучайной двоичной последовательности будет $X^{11}+X^9+1$.

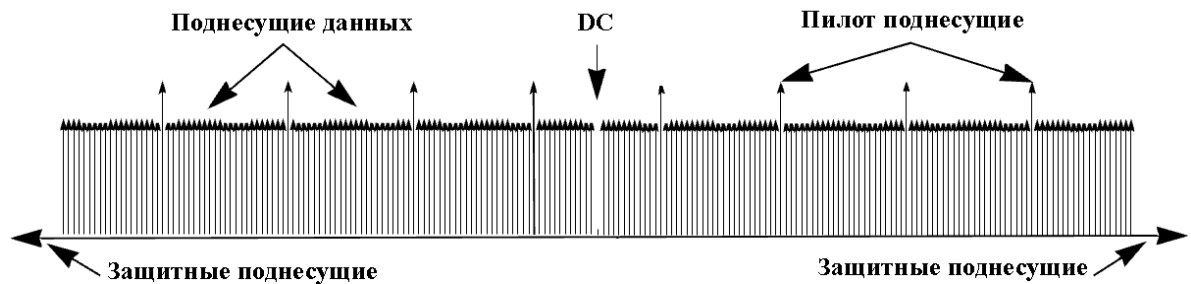


Рисунок 1.4.1 - Частотная область *OFDM* символа

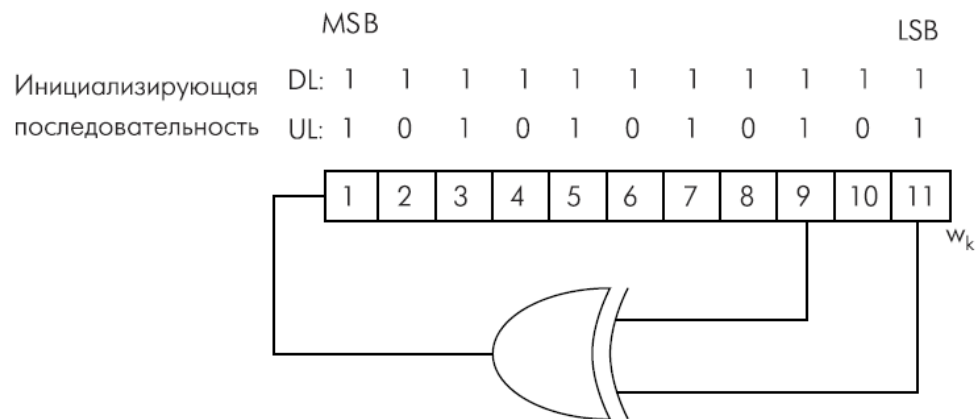


Рисунок 1.4.2 - Генерация модулирующей последовательности для пилотных поднесущих

Значение модуляции пилот-поднесущей для k -ого *OFDM* символа получается из w_k . В нисходящем канале индекс k представляет индекс символа относительно начала фрейма. В восходящем канале индекс k представляет индекс символа относительно начала барста. В обоих восходящем и нисходящем каналах первый символ преамбулы записывается с $k=0$. Последовательности инициализации, которые будут использоваться в восходящем и нисходящем каналах, показаны на рисунке 1.4.2. Для каждой пилот-поднесущей модуляция *BPSK* будет получаться как показано ниже.

Нисходящий канал (DL): $c_{-88}=c_{-38}=c_{63}=c_{88}=1-2w_k$; $c_{-88}=c_{-38}=c_{63}=c_{88}=1-2\overline{w_k}$

Восходящий канал (*UL*): $c_{-88}=c_{-38}=c_{13}=c_{38}=c_{63}=c_{88}=1-2w_k$; $c_{63}=c_{-13}=1-2\overline{w_k}$

Особенности коррекции характеристик канала по пилот-поднесущим рассмотрены в [47, 48].

Выводы по главе 1

1. В *OFDM* комплексная передаточная характеристика используется для компенсации искажений. Без такой компенсации система связи становится неработоспособной вследствие невозможности произвести правильную демодуляцию.
2. Показано, что технология *OFDM* получила широкое распространение в настоящее время за счет возможности эффективно бороться с межсимвольной интерференцией.
3. Показано, что для работы канала связи *OFDM* необходимо произвести временную, тактовую и частотную синхронизацию.
4. Рассмотрены методы временной синхронизации. Отличительной особенностью известных методов временной синхронизации является использование преамбулы с повторяющимся несколько раз сигналом во временной области.
5. Показано, что алгоритмы временной синхронизации используют решающую функцию, имеющую достаточно широкий главный максимум.
6. Показано, что широкий главный максимум решающей функции используемой для временной синхронизации приводит к снижению помехоустойчивости временной синхронизации. Таким образом, существует задача разработать такой алгоритм синхронизации, который будет иметь решающую функцию с максимально узким максимумом.
7. Показано, что эквалайзер по преамбуле наименее требователен к ресурсам, т.к. расчет коэффициентов эквалайзера производится один раз. Для расчета коэффициентов эквалайзера используется интерполяция, т.к. преамбула содержит не весь набор частотных компонент. Использование преамбулы содержащей весь набор компонент невозможно из-за

требования стандартных алгоритмов синхронизации использующих повторяющиеся участки преамбулы.

8. Показано, что если в канале имеются быстрые замирания, коррекции комплексной передаточной характеристики по преамбуле недостаточно, необходимо комбинировать эквалайзер по преамбуле с эквалайзером по пилот-поднесущим.
9. Рассмотрено расположение пилот-поднесущих в соответствии со стандартом *WiMAX OFDM* 802.16-2004.
10. Показано, что технология *DMT*, отличается от *OFDM* только симметричным относительно $N/2$ ($f_0/2$) спектром. Сигнал во временной области содержит только реальную часть. Технология *DMT* начинает использоваться в оптических средах передачи для достижения скорости более 100 Гбит/с.

Глава 2. Развитие методов синхронизации и коррекции комплексной передаточной характеристики системы связи по преамбуле и пилот-поднесущим

В параграфе 2.1 предложена реализация алгоритма коррекции КПХ с использованием в расчетах полярной системы координат, т.е. в соответствии со стандартом *WiMAX 802.16.2004*.

В параграфе 2.2 предложена и обоснована модификация типового алгоритма коррекции КПХ, позволяющая отказаться от работы в полярной системе координат, что упрощает реализацию алгоритма.

В параграфе 2.3 представлен целочисленный алгоритм коррекции КПХ системы связи, использующий в расчетах ортогональную систему координат.

В параграфе 2.4 предложен алгоритм расчета оптимальной по пик-фактору преамбулы для модифицированного алгоритма коррекции КПХ.

В параграфе 2.5 предложен способ временной синхронизации, использующий априорную информацию о преамбуле, основанный на взаимной корреляции принятого сигнала и эталонной преамбулы.

В параграфе 2.6 рассмотрен алгоритм тактовой и частотной синхронизации.

2.1. Алгоритм коррекции КПХ системы связи с применением в расчетах полярной системы координат

Методика определения комплексной передаточной характеристики с использованием линейчатого спектра основана на применении полигармонических шумоподобных сигналов в качестве испытательной последовательности. КПХ строится по известным значениям комплексных коэффициентов передачи для различных значений частот гармонических составляющих. Определение КПХ дает возможность произвести компенсацию искажений сигнала в системе связи, т.е. привести спектр сигнала в требуемый диапазон по амплитуде и подстроить фазовые набег.

Компенсация искажений осуществляется путем комплексного умножения обратной КПХ на спектр каждого последующего символа. От того насколько точно рассчитана обратная КПХ напрямую зависит правильность демодуляции принятого сигнала. Чем точнее проведена коррекция обратной комплексной передаточной характеристикой, тем меньше ошибочных данных будет получено при демодуляции.

Алгоритм коррекции КПХ системы связи с применением в расчетах полярной системы координат использует преамбулу в соответствии со стандартом *WiMAX 802.16.2004*, которая состоит из двух *OFDM* символов (см. рисунок 2.1.1). Первый символ включает каждую четвертую компоненту и используется для синхронизации. Второй символ включает каждую вторую компоненту, например, 2-ую, 4-ую, 6-ую и т.д. и используется для синхронизации, а также для расчета обратной КПХ. При этом все нечетные компоненты неизвестны и их нужно вычислять, используя, например, линейную аппроксимацию. Именно для проведения аппроксимации и необходимо использовать полярную систему координат в расчетах, т.к. данный способ для определения неизвестных значений обратной КПХ аппроксимирует амплитуду и фазу, чтобы потом получить неизвестные значения реальной и мнимой частей каждой точки массива комплексного коэффициента передачи.

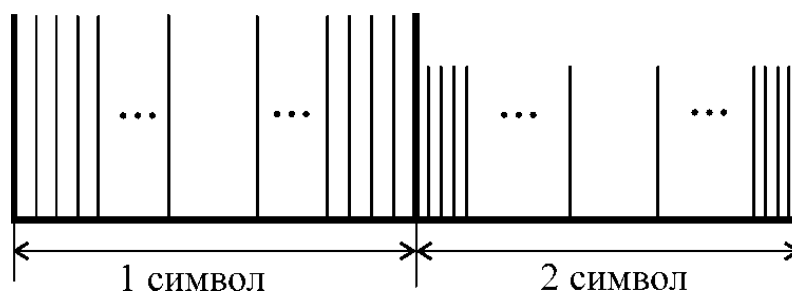


Рисунок 2.1.1 - Спектр преамбулы *WiMAX 802.16.2004*

Алгоритм коррекции КПХ системы связи, использующий стандартную преамбулу *WiMAX 802.16.2004* можно представить в виде шагов [49]:

1. Получение временных массивов (реальной и мнимой частей) преамбулы в точке, когда все спектральные компоненты сигнала

дошли до приемника. (Например, эта точка соответствует середине циклического префикса).

2. Расчет прямого преобразования Фурье от полученных массивов данных преамбулы.
3. Для каждой спектральной составляющей сигнала по формуле $K[i] = S_{IN}[i] / S_{OUT}[i]$ рассчитываются значения ОКПХ, где N – количество точек в массиве данных преобразования Фурье, S_{IN} – массив спектральных составляющих преамбулы (эталонный сигнал), S_{OUT} – массив спектральных составляющих преамбулы, прошедших через канал передачи (принятый сигнал), $i = 2, 4, 6, 8$ и т.д. до $N-2$. Путем комплексного деления получаем реальную и мнимую части ОКПХ $K_{Re}[i]$ и $K_{Im}[i]$.
4. Перевод реальной и мнимой частей массива обратного комплексного коэффициента передачи $K_{Re}[i]$ и $K_{Im}[i]$ в полярные координаты, т.е. нахождение аргумента (r_K) и фазы (φ_K) [50]:

$$r_K[i] = \sqrt{K_{Re}^2[i] + K_{Im}^2[i]},$$

$$j_K[i] = \arctg\left(\frac{K_{Im}[i]}{K_{Re}[i]}\right).$$

K_{Re} – реальная часть ОКПХ, K_{Im} – мнимая часть ОКПХ, $i = 2, 4, 6, 8$ и т.д. до $N-2$.

Для внутренних точек II четверти ($K_{Re}[i] < 0$ и $K_{Im}[i] > 0$) получаем $\varphi_K[i] = \varphi_K[i] + \pi$, для внутренних точек III четверти ($K_{Re}[i] < 0$ и $K_{Im}[i] < 0$) получаем $\varphi_K[i] = \varphi_K[i] - \pi$. [51]

5. Расчет коэффициентов k и b для линейной аппроксимации ($kx+b$) аргумента (r_K) и фазы (φ_K) по формулам:

Если $\varphi_K[i+2] < \varphi_K[i]$, тогда $\varphi_K[i+2] = \varphi_K[i+2] + 2\pi$;

$k_r[i] = (r_K[i+2] - r_K[i]) / 2$, $b_r[i] = r_K[i] - k_r[i] * i$ – для аппроксимации аргумента.

$k_\varphi[i] = (\varphi_K[i+2] - \varphi_K[i]) / 2$, $b_\varphi[i] = \varphi_K[i] - k_\varphi[i] * i$ – для аппроксимации фазы.

6. Аппроксимация для получения неизвестных значений аргумента и фазы ОКПХ по формулам: $r_K[i+1]=k_r[i]*(i+1)+b_r[i]$ и $\varphi_K[i+1]=k_\varphi[i]*(i+1)+b_\varphi[i]$.
7. Шаги 5 и 6 повторяются для значений $i = 2, 4, 6, 8$ и т.д. до 98 и $i = 156, 158, 160$ и т.д. до $N-2$. Для получения неизвестных значений аргумента и фазы ОККП для $i=1$ используются коэффициенты $k_r[i]$ и $k_\varphi[i]$ рассчитанные для $i=2$. Для получения неизвестных значений аргумента и фазы ОККП для $i=255$ используются коэффициенты $k_r[i]$ и $k_\varphi[i]$ рассчитанные для $i=252$. Значения аргумента и фазы для $i \in [101, 155]$ равны нулю.
8. Перевод аргумента и фазы в декартовы координаты для получения реальной и мнимой частей ОКПХ: $K_{Re}[i]=r_K[i]*\cos(\varphi_K[i])$, $K_{Im}[i]=r_K[i]*\sin(\varphi_K[i])$, i – целое число от 1 до $N-1$.
9. Далее для подстройки по пилот-поднесущим осуществляется получение временных массивов данных (реальной и мнимой частей) модулированного определенным видом модуляции (QPSK, QAM16, QAM64) OFDM символа в точке, когда все спектральные компоненты сигнала дошли до приемника. (Например, эта точка соответствует середине циклического префикса).
10. Расчет прямого преобразования Фурье от полученных в п.9 массивов данных и получение массивов $S_DATA_Re_{OUT}[i]$ и $S_DATA_Im_{OUT}[i]$, i – целое число от 1 до $N-1$.
11. Комплексное умножение массивов $S_DATA_Re_{OUT}[i]$ и $S_DATA_Im_{OUT}[i]$ на ОКПХ ($K_{Re}[i]$ и $K_{Im}[i]$) для получения скорректированных спектров, i – целое число от 1 до $N-1$.

$$S^*_DATA_Re_{OUT}[i]=S_DATA_Re_{OUT}[i]*K_{Re}[i]-S_DATA_Im_{OUT}[i]*K_{Im}[i],$$

$$S^*_DATA_Im_{OUT}[i]=S_DATA_Re_{OUT}[i]*K_{Im}[i]+S_DATA_Im_{OUT}[i]*K_{Re}[i].$$
12. Заполнение массивов пилот-поднесущих (индексы компонент: 13, 38, 63, 88, 168, 193, 218, 243) из массивов символа $S^*_DATA_Re_{OUT}[i]$ и

$S^*_DATA_Im_{OUT}[i]$. Массивы пилот-поднесущих $Pilot_{Re}[i]$ и $Pilot_{Im}[i]$ состоят из 256 компонент, оставшиеся 248 будут найдены посредством аппроксимации.

13. Перевод реальной и мнимой частей массива пилот-поднесущих в полярные координаты, т.е. нахождение аргумента ($PilotAmpl$) и фазы ($PilotPhase$) пилот-поднесущих:

$$PilotAmpl[i] = \sqrt{Pilot_{Re}^2[i] + Pilot_{Im}^2[i]},$$

$$PilotPhase[i] = \arctg\left(\frac{Pilot_{Im}[i]}{Pilot_{Re}[i]}\right).$$

Для внутренних точек II четверти ($Pilot_{Re}[i]<0$ и $Pilot_{Im}[i]>0$) получаем $PilotPhase[i] = PilotPhase[i] + \pi$, для внутренних точек III четверти ($Pilot_{Re}[i]<0$ и $Pilot_{Im}[i]<0$) получаем $PilotPhase[i] = PilotPhase[i] - \pi$, $i=13, 38, 63, 88, 168, 193, 218, 243$.

14. Коррекция аргумента и фазы пилот-поднесущих путем вычитания известного идеального значения фазы пилот-поднесущей из значения, полученного в п.13, и деления аргумента пилот-поднесущей, полученного в п.13, на известное значение идеального аргумента:

$$kPilotAmpl[i] = PilotAmpl[i] / PilotAmplIdeal[i],$$

$$\Delta PilotPhase[i] = PilotPhase[i] - PilotPhaseIdeal[i],$$

$i=13, 38, 63, 88, 168, 193, 218, 243$.

15. Расчет коэффициентов k и b для линейной аппроксимации ($kx+b$) корректирующих аргумента ($kPilotAmpl$) и фазы ($\Delta PilotPhase$) по формулам:

$$k_A[i] = (kPilotAmpl[i+25] - kPilotAmpl[i])/25,$$

$$b_A[i] = kPilotAmpl[i] - k_A[i]*i - \text{для аппроксимации аргумента.}$$

Если $\Delta PilotPhase[i+25] < \Delta PilotPhase[i]$, тогда $\Delta PilotPhase[i+25] = \Delta PilotPhase[i+25] + 2\pi$;

$$k_{Ph}[i] = (\Delta PilotPhase[i+25] - \Delta PilotPhase[i])/25,$$

$$b_{Ph}[i] = \Delta PilotPhase[i] - k_{Ph}[i]*i - \text{для аппроксимации фазы.}$$

Расчет коэффициентов осуществляется для значений $i=13, 38, 63, 168, 193, 218$.

16. Аппроксимация для получения неизвестных значений корректирующих аргумента и фаз пилот-поднесущих по формулам:
 $kPilotAmpl[i+j]=k_A[i]*(i+j)+b_A[i]$ и $\Delta PilotPhase[i+j] = k_{Ph}[i]*(i+j)+b_{Ph}[i]$.
 Для аппроксимации шаг изменения i, j равен 1,
 $i \in [13, 63] \cup [168, 218], j \in [1, 24]$.

Для аппроксимации краевых значений, т.е. когда:

$i < 13$ и $i < 168$, используются коэффициенты $k_A[i]$ и $b_A[i]$, $k_{Ph}[i]$ и $b_{Ph}[i]$, рассчитанные для индексов $i=13$ и $i=168$ соответственно.

$kPilotAmpl[i+j]=k_A[i]*(i+j)+b_A[i]$ и $\Delta PilotPhase[i+j] = k_{Ph}[i]*(i+j)+b_{Ph}[i]$.
 $i=1$ ($k_A[13]$, $b_A[13]$ и $k_{Ph}[13]$, $b_{Ph}[13]$), 156 ($k_A[168]$, $b_A[168]$ и $k_{Ph}[168]$, $b_{Ph}[168]$), шаг изменения j равен 1, $j \in [0, 11]$.

Для аппроксимации краевых значений, т.е. когда:

$i > 63$ и $i > 218$, используются коэффициенты $k_A[i]$ и $b_A[i]$, $k_{Ph}[i]$ и $b_{Ph}[i]$, рассчитанные для индексов $i=63$ и $i=218$ соответственно.

$kPilotAmpl[i+j]=k_A[i]*(i+j)+b_A[i]$ и $\Delta PilotPhase[i+j] = k_{Ph}[i]*(i+j)+b_{Ph}[i]$.
 $i=89$ ($k_A[63]$, $b_A[63]$ и $k_{Ph}[63]$, $b_{Ph}[63]$), 244 ($k_A[218]$, $b_A[218]$ и $k_{Ph}[218]$, $b_{Ph}[218]$), шаг изменения j равен 1, $j \in [0, 11]$.

Массивы $kPilotAmpl[i]$ и $\Delta PilotPhase[i]$ равны нулю при i от 101 до 155.

17. Нахождение аргументов и фаз $SignalAmpl[i]$, $SignalPhase[i]$ компонент скорректированных обратным комплексным коэффициентом передачи массивов символа данных, где i – целое число от 1 до $N-1$.

$$SignalAmpl[i] = \sqrt{S * _DATA_Re_{OUT}^2[i] + S * _DATA_Im_{OUT}^2[i]},$$

$$SignalPhase[i] = \arctg \left(\frac{S * _DATA_Im_{OUT}^2[i]}{S * _DATA_Re_{OUT}^2[i]} \right).$$

18. Коррекция аргумента и фазы массивов данных корректирующими аргументами и фазами пилот-поднесущих:

$$kSignalAmpl[i] = SignalAmpl[i] / kPilotAmpl[i],$$

$\Delta SignalPhase[i] = SignalPhase[i] - \Delta PilotPhase[i]$, где i – целое число от 1 до $N-1$.

19. Перевод скорректированных пилот-поднесущими аргумента и фазы сигнала в декартовы координаты для получения реальной и мнимой частей компонент сигнала:

$$Signal_{Re}[i] = kSignalAmpl[i] * \cos(\Delta SignalPhase[i]),$$

$$Signal_{Im}[i] = kSignalAmpl[i] * \sin(\Delta SignalPhase[i]),$$
 где i – целое число от 1 до $N-1$.

2.2. Алгоритм коррекции КПХ системы связи с использованием в расчетах ортогональной системы координат

Для упрощения реализации на аппаратном уровне алгоритмов работы *OFDM* системы связи и снижения требований к скорости выполнения операции деления предлагается алгоритм коррекции КПХ на основе разделения преамбулы на 2 символа и исключения из алгоритма процедур перевода в полярную систему координат и обратно. Следует иметь в виду, что математические действия при переходе в полярную систему координат ресурсоемки (требуют вычисления квадратного корня и арктангенса для каждой точки в массиве ОККП).

Типовая процедура определения массива комплексного коэффициента передачи (ККП) в системе связи *OFDM* предполагает передачу в преамбуле известного испытательного линейчатого спектра, по искажениям которого осуществляется коррекция комплексной передаточной характеристики системы связи. Поскольку амплитудные и фазовые искажения спектра сигнала во многих случаях обусловлены относительно медленными изменениями КПХ, то возможна оценка массива ККП в интервале передачи не одного, а нескольких символов с разделением во времени частот испытательного линейчатого спектра. [52]

При передаче преамбулы в N символах уменьшается количество спектральных составляющих в каждом символе. Например, если передавать преамбулу в двух символах, то число спектральных компонент в каждом из них будет уменьшено в два раза. Тогда при сохранении мощности уровень этих компонент поднимается в $\sqrt{2}$ раз. Для того же уровня шума отношение С/Ш тоже возрастет в $\sqrt{2}$ раз, поэтому погрешность расчета ККП уменьшится. [53,54]

Модифицированный алгоритм коррекции КПХ предполагает использование двух *OFDM* символов в преамбуле. В первом символе преамбулы передаются компоненты с 0 по 50 и с 206 по 255. Во втором с 51 по 100 и с 156 по 205 (см. рисунок 2.2.1). [55]

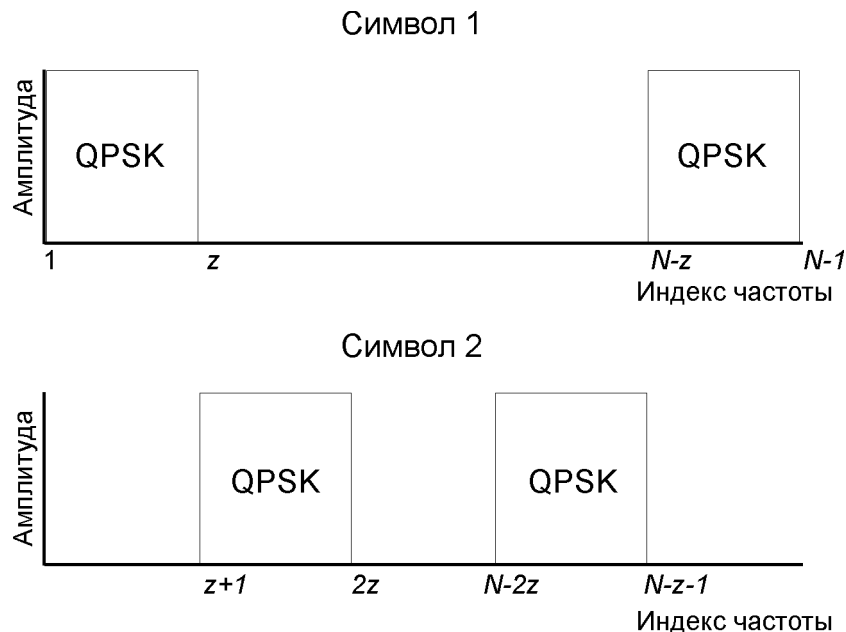


Рисунок 2.2.1 - Условный амплитудный спектр преамбулы для предложенного алгоритма

Количество спектральных компонент в одном символе равно $2z$, $2z=100$. Количество точек в преобразовании Фурье $N=256$.

Модифицированный алгоритм коррекции КПХ:

1. Получение временных массивов (реальной и мнимой частей) преамбулы в точке, когда все спектральные компоненты сигнала дошли до приемника. (Например, эта точка соответствует середине циклического префикса).
2. Расчет прямого преобразования Фурье от полученных массивов данных.

3. Для каждой спектральной составляющей сигнала по формуле $K[i] = S_{IN}[i]/S_{OUT}[i]$ рассчитываются значения ОКП, где N – количество точек в массиве данных преобразования Фурье, S_{IN} – массив спектральных составляющих преамбулы (эталонный сигнал), S_{OUT} – массив спектральных составляющих преамбулы, прошедших через канал передачи (принятый сигнал), i – целое число от 1 до $N-1$. Получение реальной и мнимой частей ОКП $K_{Re}[i]$ и $K_{Im}[i]$.
4. Далее для подстройки по пилот-поднесущим осуществляется получение временных массивов данных модулированного определенным видом модуляции ($QPSK$, $QAM16$, $QAM64$) $OFDM$ символа в точке, когда все спектральные компоненты сигнала дошли до приемника. (Например, эта точка соответствует середине циклического префикса).
5. Расчет прямого преобразования Фурье от полученных массивов данных и получение массивов $S_DATA_Re_{OUT}[i]$ и $S_DATA_Im_{OUT}[i]$, i – целое число от 1 до $N-1$.
6. Умножение массивов $S_DATA_Re_{OUT}[i]$ и $S_DATA_Im_{OUT}[i]$ на значения ОКП ($K_{Re}[i]$ и $K_{Im}[i]$), полученные в п.3 в комплексном виде для получения скорректированных спектров данных, i – целое число от 1 до $N-1$.

$$S^*_DATA_Re_{OUT}[i] = S_DATA_Re_{OUT}[i] * K_{Re}[i] - S_DATA_Im_{OUT}[i] * K_{Im}[i],$$

$$S^*_DATA_Im_{OUT}[i] = S_DATA_Re_{OUT}[i] * K_{Im}[i] + S_DATA_Im_{OUT}[i] * K_{Re}[i].$$
7. Заполнение массивов пилот-поднесущих $Pilot_{Re}[i]$, $Pilot_{Im}[i]$ (индексы компонент: 2, 34, 66, 98, 158, 190, 222, 254) из массивов скорректированных обратным комплексным коэффициентом передачи спектров данных $S^*_DATA_Re_{OUT}[i]$ и $S^*_DATA_Im_{OUT}[i]$. Массивы пилот-поднесущих состоят из 256 компонент, оставшиеся 248 будут найдены посредством аппроксимации.
8. Расчет значений ОКП по пилот-поднесущим:

$$OKKPPilot_{Re}[i] = PilotIdeal_{Re}[i] / Pilot_{Re}[i],$$

$$OKKPPilot_{Im}[i] = PilotIdeal_{Im}[i] / Pilot_{Im}[i],$$

$i=2, 34, 66, 98, 158, 190, 222, 254$.

9. Расчет коэффициентов k и b для линейной аппроксимации $(kx+b)$ ОКПХ, для реальной и мнимой частей:

$$k_{Re}[i] = (OKKPPilot_{Re}[i+32] - OKKPPilot_{Re}[i])/32;$$

$$b_{Re}[i] = OKKPPilot[i] - k_{Re}[i]*i.$$

$$k_{Im}[i] = (OKKPPilot_{Im}[i+32] - OKKPPilot_{Im}[i])/32;$$

$$b_{Im}[i] = OKKPPilot[i] - k_{Im}[i]*i.$$

Расчет коэффициентов осуществляется для значений $i=2, 34, 66, 158, 190, 222$.

10. Аппроксимация для получения неизвестных значений реальной и мнимой частей ОКПХ по формуле: $OKKPPilot_{Re}[i+j] = k_{Re}[i]*(i+j) + b_{Re}[i]$, $OKKPPilot_{Im}[i+j] = k_{Im}[i]*(i+j) + b_{Im}[i]$.

Для аппроксимации шаг изменения i, j равен 1, $i \in [2, 66] \cup [158, 222], j \in [1, 31]$.

Для аппроксимации краевых значений, т.е. когда:

$i < 2$ и $i < 158$, используются коэффициенты $k[i]$ и $b[i]$, рассчитанные для индексов $i=2$ и $i=158$ соответственно для реальной и мнимой частей.

$$OKKPPilot_{Re}[i+j] = k_{Re}[i]*(i+j) + b_{Re}[i], \quad OKKPPilot_{Im}[i+j] = k_{Im}[i]*(i+j) + b_{Im}[i],$$

где $i=0$ ($k[2]$ и $b[2]$), 156 ($k[158]$ и $b[158]$), $j=0, 1$.

Для аппроксимации краевых значений, т.е. когда:

$i > 66$ и $i > 222$, используются коэффициенты k и b , рассчитанные для индексов $i=66$ и $i=222$ соответственно для реальной и мнимой частей.

$$OKKPPilot_{Re}[i+j] = k_{Re}[i]*(i+j) + b_{Re}[i], \quad OKKPPilot_{Im}[i+j] = k_{Im}[i]*(i+j) + b_{Im}[i],$$

где $i=99$ ($k[66]$ и $b[66]$), 255 ($k[158]$ и $b[158]$), где $j=0, 1$ для $i=99$, $j=0$ для $i=255$.

Массивы $OKKPPilot_{Re}[i]$ и $OKKPPilot_{Im}[i]$ равны нулю при i от 101 до 155.

11. Комплексное умножение частей ОКПХ, рассчитанных по пилот-поднесущим, $OKKPPilot_{Re}[i]$ и $OKKPPilot_{Im}[i]$ на массивы данных $S*_{DATA_Re_OUT}[i]$ и $S*_{DATA_Im_OUT}[i]$ для коррекции:

$$Signal_{Re}[i] = S_DATA_Re_{OUT}[i] * OKKPPilot_{Re}[i] - S_DATA_Im_{OUT}[i] * OKKPPilot_{Im}[i].$$

$$Signal_{Im}[i] = S_DATA_Re_{OUT}[i] * OKKPPilot_{Im}[i] + S_DATA_Im_{OUT}[i] * OKKPPilot_{Re}[i].$$

Где i – целое число от 1 до $N-1$.

Использование модифицированного алгоритма позволило:

- сократить количество шагов алгоритма на 8 позиций;
- отказаться от перевода из ортогональной системы координат в полярную систему координат;
- отказаться от перевода из полярной системы координат в ортогональную систему координат;
- отказаться от использования аппроксимации при расчете обратной комплексной передаточной характеристики по преамбуле.

2.3. Целочисленный алгоритм коррекции КПХ системы связи с использованием в расчетах ортогональной системы координат

Предложенный алгоритм работает с вещественными значениями, что неприемлемо для его реализации в аппаратуре, т.к. при реализации в аппаратуре одной из самых медленных операций является операция деления, чем большая битность используется в модуле деления, тем большее количество тактов потребуется для выполнения деления. Максимально просто на языке описания аппаратуры реализуется операция целочисленного деления, однако ее использование без модификации алгоритма недопустимо, т.к. произойдет неприемлемое падение точности. Для реализации алгоритма коррекции с использованием целочисленного деления необходимо найти нормирующие коэффициенты, которые позволят сохранить точность расчета ОКПХ. [56,57,58]

Целочисленный алгоритм коррекции КПХ по преамбуле и пилот-поднесущим, модулированным по схеме *QPSK*:

1) Получение с АЦП временных массивов (реальной - $X_{Re}[i]$ и мнимой - $X_{Im}[i]$ частей) преамбулы в диапазоне оптимальных значений ± 96 для 8ми

битного АЦП, ± 390 для 10ти битного АЦП и ± 1535 для 12ти битного АЦП, в середине циклического префикса, i – целое число от 1 до $N-1$, N – количество точек в массиве данных преобразования Фурье.

2) Расчет прямого преобразования Фурье от полученных массивов данных $X_{Re}[i]$, $X_{Im}[i]$. Получение реальной – $S_{Re}[i]$ и мнимой – $S_{Im}[i]$ частей спектра принятой преамбулы по формуле:

$$S[i] = FFT(X_{Re}[i], X_{Im}[i]),$$

где $S[i]$ - комплексное значение спектра принятой преамбулы в точке i , i – целое число от 1 до $N-1$.

Получение массива спектральных значений $S_{Re}[i]$, $S_{Im}[i]$ в диапазоне ± 26000 для 8, 10, 12-ти битного АЦП:

$$S_{Re}[i] = S_{Re}[i] \ll M,$$

$$S_{Im}[i] = S_{Im}[i] \ll M,$$

где операция $\ll M$ означает сдвиг влево на M бит, $M = 4$, если используется 8ми битный АЦП, $M = 2$, если используется 10ти битный АЦП, $M = 0$, если используется 12ти битный АЦП, i – целое число от 1 до $N-1$.

3) Для каждой спектральной составляющей преамбулы рассчитываются значения ОККП по формуле:

$$K[i] = S_{IN}[i] / S[i],$$

где $K[i]$ - массив ОККП, $S_{IN}[i]$ – комплексный массив спектральных составляющих преамбулы (эталонная преамбула), $S[i]$ – комплексный массив спектральных составляющих преамбулы, прошедших через канал передачи (принятая преамбула), i – целое число от 1 до $N-1$.

Расчет ОККП производится по правилу деления комплексных чисел. Для расчета знаменателя значения реальной и мнимой частей спектра $S_{Re}[i]$, $S_{Im}[i]$ сдвигаются вправо на 1 бит ($>> 1$) для получения $S_{Re_}[i]$, $S_{Im_}[i]$:

$$S_{Re_}[i] = S_{Re}[i] \gg 1,$$

$$S_{Im_}[i] = S_{Im}[i] \gg 1,$$

где i – целое число от 1 до $N-1$.

Расчет знаменателя (z) осуществляется по формуле:

$$z[i] = (S_{Re_}[i]^2 + S_{Im_}[i]^2) \gg 15.$$

Получение реальной и мнимой частей ОКПХ $K_{Re}[i]$ и $K_{Im}[i]$ в диапазоне значений ± 29000 для 8, 10, 12-ти битного АЦП по формулам:

$$K_{Re}[i] = (S_{IN_Re}[i] \cdot S_{Re}[i] + S_{IN_Im}[i] \cdot S_{Im}[i]) / z[i],$$

$$K_{Im}[i] = (S_{IN_Im}[i] \cdot S_{Re}[i] - S_{IN_Re}[i] \cdot S_{Im}[i]) / z[i],$$

где $S_{IN_Re}[i]$, $S_{IN_Im}[i]$ – реальная и мнимая части спектральных составляющих эталонной преамбулы, i – целое число от 1 до $N-1$.

4) Далее осуществляется получение временных массивов данных ($X_{Sig_Re}[i]$, $X_{Sig_Im}[i]$) модулированного определенным видом модуляции (QPSK, QAM16, QAM64) OFDM символа в середине циклического префикса. Оптимальные значения массивов $X_{Sig_Re}[i]$, $X_{Sig_Im}[i]$ располагаются в диапазоне ± 96 для 8ми битного АЦП, ± 390 для 10ти битного АЦП и ± 1535 для 12ти битного АЦП, i – целое число от 1 до $N-1$.

5) Расчет прямого преобразования Фурье от полученных массивов данных $X_{Sig_Re}[i]$, $X_{Sig_Im}[i]$. Получение реальной – $S_{Sig_Re}[i]$ и мнимой – $S_{Sig_Im}[i]$ частей спектра принятого сигнала по формуле:

$$S_{Sig}[i] = FFT(X_{Sig_Re}[i], X_{Sig_Im}[i]),$$

где $S_{Sig}[i]$ – комплексное значение спектра принятого сигнала в точке i , i – целое число от 1 до $N-1$.

Сдвиг реальной и мнимой частей спектра ($S_{Sig_Re}[i]$, $S_{Sig_Im}[i]$) влево на 4 бита ($\ll 4$) в случае использования 8ми битного АЦП, на 2 бита ($\ll 2$) в случае использования 10ти битного АЦП и 0 бит ($\ll 0$) в случае использования 12ти битного АЦП для получения значений $S_{Sig_Re}[i]$, $S_{Sig_Im}[i]$ в диапазоне ± 26500 для 8, 10, 12-ти битного АЦП:

$$S_{Sig_Re_}[i] = S_{Sig_Re}[i] \ll M,$$

$$S_{Sig_Im_}[i] = S_{Sig_Im}[i] \ll M,$$

где $M = 4$, если используется 8ми битный АЦП, $M = 2$, если используется 10ти битный АЦП, $M = 0$, если используется 12ти битный АЦП, операция $\ll M$ означает сдвиг влево на M бит, i – целое число от 1 до $N-1$.

6) Применение ОКПХ ($K_{Re}[i]$ и $K_{Im}[i]$) к массивам данных $S_{Sig_Re}[i]$ и $S_{Sig_Im}[i]$ в комплексном виде для получения скорректированных по преамбуле спектров символа данных и сдвиг их вправо на 15 бит ($\gg 15$) для получения значений скорректированных спектров в диапазоне ± 16000 :

$$S_{Sig_Re_OKKP}[i] = (S_{Sig_Re}[i] * K_{Re}[i] - S_{Sig_Im}[i] * K_{Im}[i]) \gg 15,$$

$$S_{Sig_Im_OKKP}[i] = (S_{Sig_Re}[i] * K_{Im}[i] + S_{Sig_Im}[i] * K_{Re}[i]) \gg 15, \quad i - \text{целое число от 1 до } N-1.$$

7) Заполнение массивов пилот-поднесущих $Pilot_{Re}[i]$, $Pilot_{Im}[i]$ (индексы компонент: 2, 34, 66, 98, 158, 190, 222, 254) из массивов скорректированных обратным комплексным коэффициентом передачи спектров данных $S_{Sig_Re_OKKP}[i]$ и $S_{Sig_Im_OKKP}[i]$.

8) Расчет значений ОКПХ по пилот-поднесущим осуществляется по формуле:

$$OKKP_{pilot}[i] = Pilot_{IN}[i] / Pilot[i],$$

где $OKKP_{pilot}[i]$ - массив ОККП, рассчитанный по пилот-поднесущим,

$Pilot_{IN}[i]$ – комплексный массив пилот-поднесущих (эталонные значения пилот-поднесущих), $Pilot[i]$ – комплексный массив пилот-поднесущих, прошедших через канал передачи (принятые значения пилот-поднесущих), $i=2, 34, 66, 98, 158, 190, 222, 254$.

Расчет ОККП производится по правилу деления комплексных чисел. Для расчета знаменателя значения реальной и мнимой частей пилот-поднесущих

$Pilot_{Re}[i]$, $Pilot_{Im}[i]$ сдвигаются вправо на 1 бит ($>> 1$) для получения $Pilot_{Re_}[i]$, $Pilot_{Im_}[i]$:

$$Pilot_{Re_}[i] = Pilot_{Re}[i] >> 1,$$

$$Pilot_{Im_}[i] = Pilot_{Im}[i] >> 1,$$

где $i=2, 34, 66, 98, 158, 190, 222, 254$.

Расчет знаменателя (z_{Pilot}) осуществляется по формуле:

$$Z_{Pilot}[i] = (Pilot_{Re_}[i]^2 + Pilot_{Im_}[i]^2) >> 15.$$

Получение реальной и мнимой частей ОКПХ $OKKP_{Pilot_Re}[i]$ и $OKKP_{Pilot_Im}[i]$ в диапазоне значений ± 33000 для 8, 10, 12-ти битного АЦП по формулам:

$$OKKP_{Pilot_Re}[i] = (Pilot_{IN_Re}[i] \cdot Pilot_{Re}[i] + Pilot_{IN_Im}[i] \cdot Pilot_{Im}[i]) / z_{Pilot}[i],$$

$$OKKP_{Pilot_Im}[i] = (Pilot_{IN_Im}[i] \cdot Pilot_{Re}[i] - Pilot_{IN_Re}[i] \cdot Pilot_{Im}[i]) / z_{Pilot}[i],$$

где $Pilot_{IN_Re}[i]$, $Pilot_{IN_Im}[i]$ – эталонные значения реальной и мнимой части пилот-поднесущих, где $i=2, 34, 66, 98, 158, 190, 222, 254$.

9) Расчет коэффициентов k и b для линейной аппроксимации ($kx+b$) ОКПХ, для реальной и мнимой частей:

$$k_{Re}[i] = (OKKP_{Pilot_Re}[i+32] - OKKP_{Pilot_Re}[i]) / 32;$$

$$b_{Re}[i] = OKKP_{Pilot_Re}[i] - k_{Re}[i] * i.$$

$$k_{Im}[i] = (OKKP_{Pilot_Im}[i+32] - OKKP_{Pilot_Im}[i]) / 32;$$

$$b_{Im}[i] = OKKP_{Pilot_Im}[i] - k_{Im}[i] * i.$$

Расчет коэффициентов осуществляется для значений $i=2, 34, 66, 158, 190, 222$.

10) Аппроксимация для получения неизвестных значений реальной и мнимой частей ОКПХ по формуле:

$$OKKP_{Pilot_Re}[i+j] = k_{Re}[i] * (i+j) + b_{Re}[i],$$

$$OKKP_{Pilot_Im}[i+j] = k_{Im}[i] * (i+j) + b_{Im}[i].$$

Для аппроксимации j – целое число от 1 до 31, $i=2, 34, 66, 158, 190, 222$.

Для аппроксимации краевых значений, т.е. когда:

$i < 2$ и $i < 158$, используются коэффициенты $k[i]$ и $b[i]$, рассчитанные для индексов $i=2$ и $i=158$ соответственно для реальной и мнимой частей.

$OKKP_{Pilot_Re}[i+j]=k_{Re}[i]*(i+j)+b_{Re}[i]$, $OKKP_{Pilot_Im}[i+j]=k_{Im}[i]*(i+j)+b_{Im}[i]$, где $i=0$ ($k[2]$ и $b[2]$), 156 ($k[158]$ и $b[158]$), $j=0, 1$.

Для аппроксимации краевых значений, т.е. когда:

$i>66$ и $i>222$, используются коэффициенты k и b , рассчитанные для индексов $i=66$ и $i=222$ соответственно для реальной и мнимой частей.

$$OKKP_{Pilot_Re}[i+j]=k_{Re}[i]*(i+j)+b_{Re}[i],$$

$$OKKP_{Pilot_Im}[i+j]=k_{Im}[i]*(i+j)+b_{Im}[i],$$

где $i=99$ ($k[66]$ и $b[66]$), 255 ($k[158]$ и $b[158]$), где $j=0, 1$ для $i=99$, $j=0$ для $i=255$.

Массивы $OKKP_{Pilot_Re}[i]$ и $OKKP_{Pilot_Im}[i]$ равны нулю при i от 101 до 155.

11) Комплексное умножение частей ОКПХ, рассчитанных по пилот-поднесущим, $OKKP_{Pilot_Re}[i]$ и $OKKP_{Pilot_Im}[i]$ на массивы данных $S_{Sig_Re_OKKP}[i]$ и $S_{Sig_Im_OKKP}[i]$ для коррекции:

$$Signal_{Re}[i]=S_{Sig_Re_OKKP}[i] * OKKP_{Pilot_Re}[i] - S_{Sig_Im_OKKP}[i] * OKKP_{Pilot_Im}[i],$$

$$Signal_{Im}[i]=S_{Sig_Re_OKKP}[i] * OKKP_{Pilot_Im}[i] + S_{Sig_Im_OKKP}[i] * OKKP_{Pilot_Re}[i].$$

Для избежания переполнения при 32-х битном умножении производится уменьшение значений массивов $Signal_{Re}[i]$ и $Signal_{Im}[i]$, путем сдвига их значений вправо на 14 бит:

$$Signal_{Re}[i]= Signal_{Re}[i] >> 14 ,$$

$$Signal_{Im}[i]= Signal_{Im}[i] >> 14 .$$

Восстановление исходной амплитуды спектральных компонент по известному отношению амплитуды данных к амплитуде пилот-поднесущих для получения значений реальной и мнимой частей спектра данных в диапазоне $\pm 2^{26}$ для 8, 10, 12-ти битного АЦП:

$$Signal_{Re}[i]= Signal_{Re}[i] * B,$$

$$Signal_{Im}[i]= Signal_{Im}[i] * B,$$

где $B=11070$ – нормирующий коэффициент, являющийся отношением 2^{12} к известному отношению амплитуды данных (A_D) к амплитуде пилот-поднесущих (A_P): $A_D/A_P=0,37$, т.е. получаем $2^{12}/0,37$, i – целое число от 1 до $N-1$.

Операции деления, используемые в алгоритме, являются целочисленными. Алгоритм функционирует в диапазоне минимального входного значения сигнала $\pm 2^{\text{Битность АЦП-1}}/32$, максимального входного значения сигнала $\pm 2^{\text{Битность АЦП-1}}$.

2.4. Определение оптимальной преамбулы для предложенного алгоритма коррекции КПХ системы связи

Для определения целесообразности использования компонент стандартной преамбулы в модифицированном алгоритме сравним пиковые значения сигнала в одном и другом случаях. Следует отметить, что СКЗ символов преамбул будут равны вследствие одинакового количества спектральных составляющих, которые имеют равную амплитуду. Рассмотрим временную область стандартной преамбулы в исполнении стандартного алгоритма (см. рисунок 2.4.1).

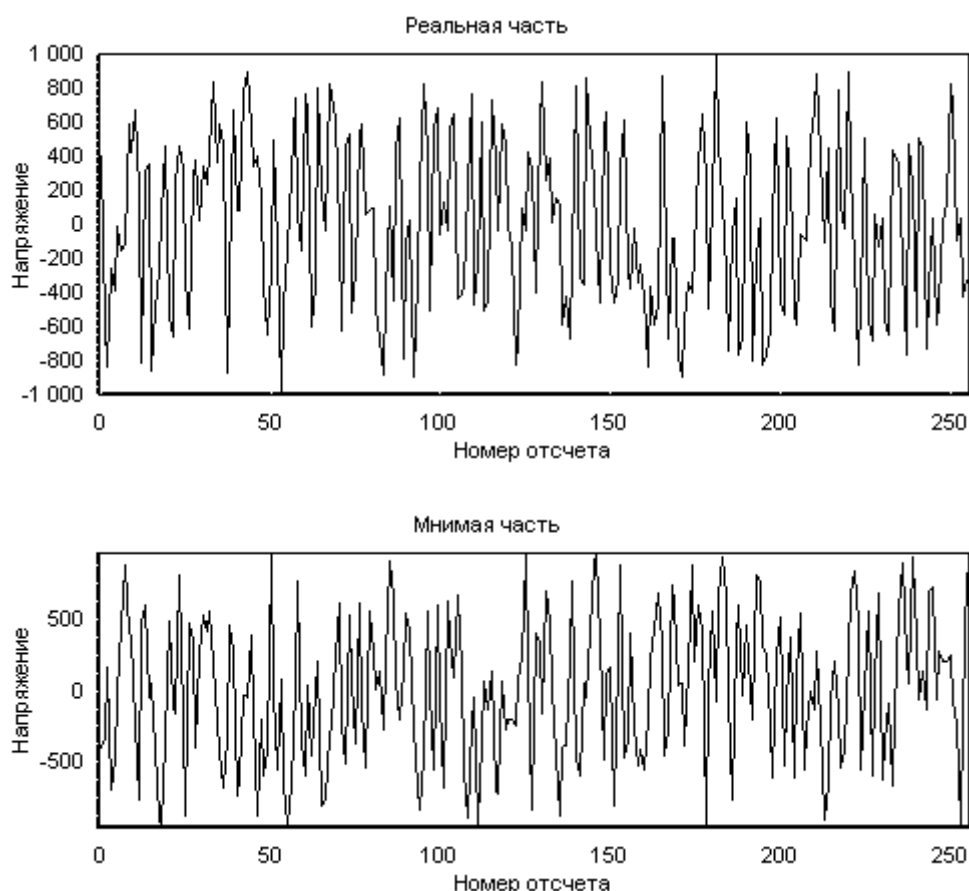


Рисунок 2.4.1 - Временная область преамбулы (реальная и мнимая части) по стандарту WiMAX

Рассмотрим временную область стандартной преамбулы для использования в модифицированном алгоритме (см. рисунки 2.4.2, 2.4.3).

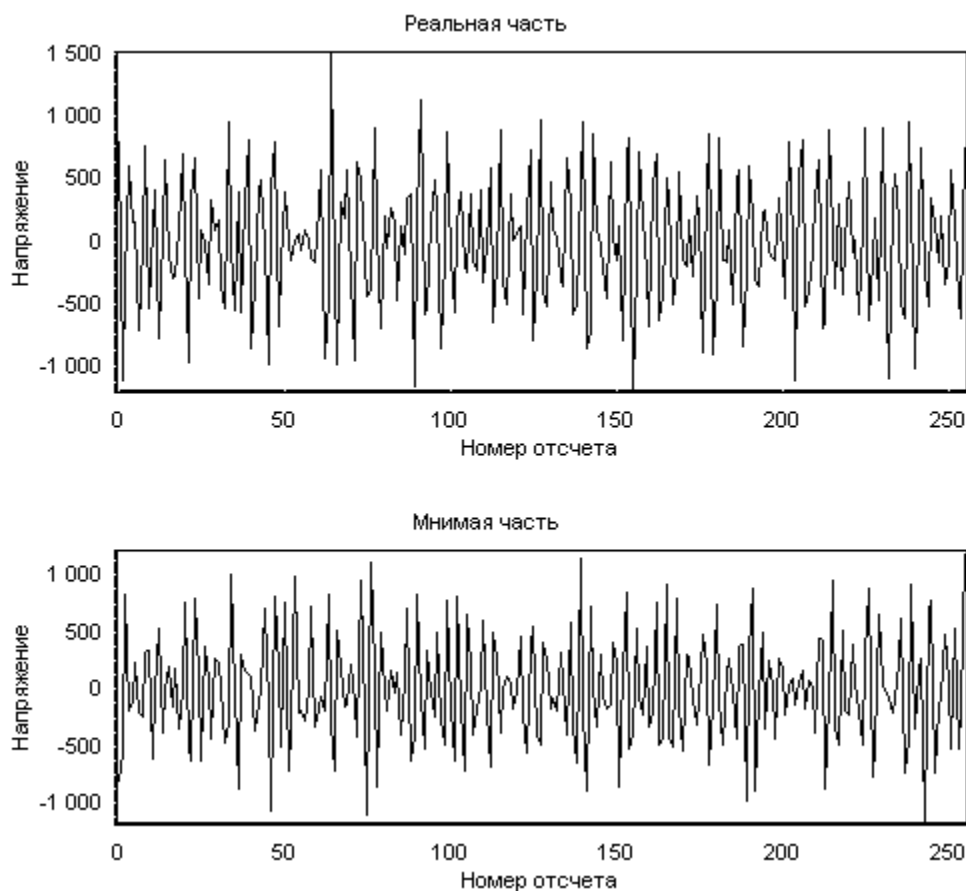


Рисунок 2.4.2 - Временная область стандартной преамбулы, символ, содержащий ВЧ компоненты

Пик-фактор для символа преамбулы, содержащего ВЧ компоненты равен 2,16. Пик-фактор для символа преамбулы, содержащего НЧ компоненты равен 2,04.

Пик-фактор определяет отношение максимального пикового значения сигнала к его среднеквадратическому значению, т.е. величине определяющей мощность. Сигнал с большим значением пик-фактора имеет малую среднюю мощность (энергию) и большое пиковое значение мощности. [59], [60] Такой режим работы является крайне неоптимальным для передатчика и приемника. Минимизация пик-фактора позволяет без изменения схемной реализации устройства повысить дальность связи и (или) увеличить помехоустойчивость за счет использования сигналов с большой энергией.

Для *OFDM* снижение пик-фактора особенно актуально из-за требования линейности. Так как для корректной работы *OFDM* необходимо, чтобы передатчик и приемник работали в линейном режиме, т.к. возникновение нелинейных искажений приведет к искажению спектра и скорректировать комплексную передаточную характеристику будет невозможно. [61]

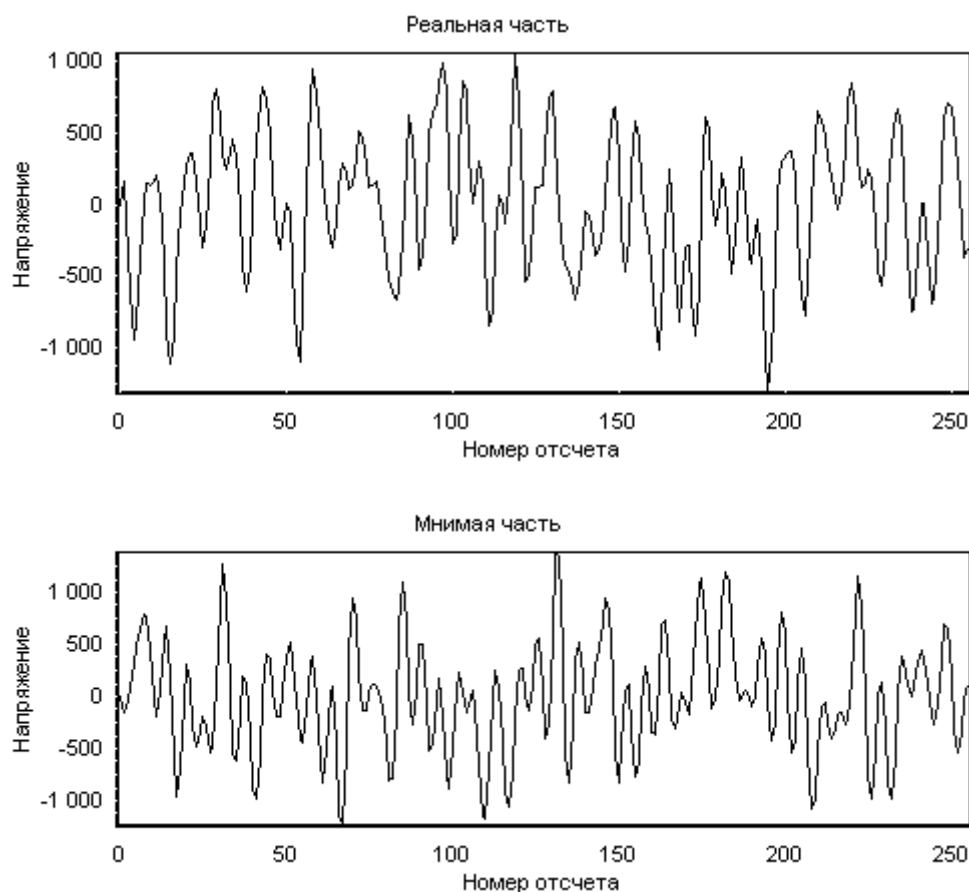


Рисунок 2.4.3 - Временная область стандартной преамбулы, символ, содержащий НЧ компоненты

Для снижения пик-фактора преамбулы можно использовать различные методы [62, 63], например, подбора фаз [64], т.е. подбора значений реальной и мнимой части спектральных составляющих *QPSK*, однако реализовать проверку всех возможных вариантов даже для размера преобразования Фурье 256 невозможно. Например, в стандарте *WiMAX* [65] *OFDM* используется 200 спектральных составляющих, из которых только 100 передаются в преамбуле, т.е. всего возможно $4^{100} \approx 1,6 \cdot 10^{60}$ различных вариантов преамбулы. Даже если для проверки пик-фактора будет

требоваться 1нс на одну итерацию это потребует приблизительно 10^{46} лет для поиска оптимального варианта спектра. Вследствие этого метод прямого перебора не применим, однако можно применить генератор случайных чисел и генерировать случайные преамбулы *QPSK* с требуемым спектральным заполнением и сохранять такую преамбулу, для которой найден минимальный пик-фактор.

В среде *Delphi* была разработана программа расчета коэффициентов преамбулы для поиска ее оптимального варианта. [66]

На рисунке 2.4.4 представлено рабочее окно программы расчета коэффициентов преамбулы, на котором приведена стандартная преамбула *WiMAX 802.16-2004 OFDM* без нормализации к диапазону АЦП. Среднеквадратическое значение (СКЗ) преамбулы равно $СКЗ = \sqrt{\frac{1}{4} \sum_{i=1}^{N-1} (S_{re}^2[i] + S_{im}^2[i])}$, где S_{re} , S_{im} - реальные и мнимые амплитудные части спектра преамбулы.

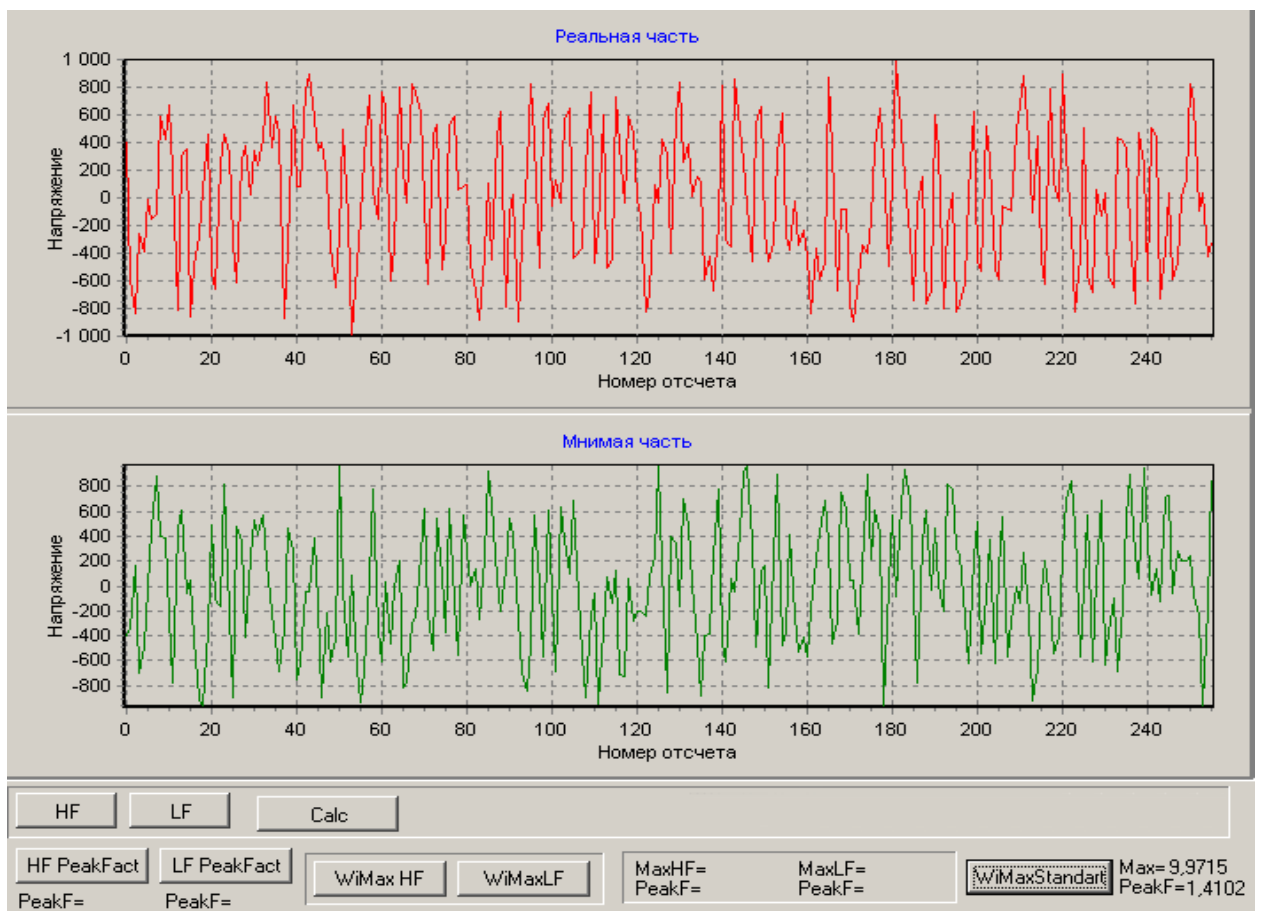


Рисунок 2.4.4. Рабочее окно программы расчета коэффициентов преамбулы

Для абсолютных значений реальных и мнимых частей спектральных составляющих равных 1 при учете, что стандартная преамбула состоит из 100 спектральных четных компонент, получим $СКЗ = \sqrt{\frac{1}{4} \cdot 100 \cdot (1 + 1)} = 7,07$. Пик-фактор для такой преамбулы равен 1,41. Однако в случае использования разделения преамбулы на высокочастотные и низкочастотные составляющие [52] пик-фактор для ВЧ части будет составлять 2,16 и 2,04 для НЧ части.

Предлагается алгоритм поиска преамбулы для минимизации пик-фактора, состоящий из следующих шагов [61]:

1. Обнуление коэффициентов массива преамбулы. $S_{re}[i]=0$, $S_{im}[i]=0$ - реальные и мнимые амплитудные части спектра преамбулы, $i \in 0..255$.
2. Установка значения $GlobalPeak=1000$. Большое значение этой переменной приведет к ее перезаписи на первой же итерации цикла поиска оптимальной преамбулы.
3. Установка счетчика количества итераций в ноль. $Cnt=0$.
4. Заполнение массивов коэффициентов преамбулы $S_{re}[i]$, $S_{im}[i]$ случайными значениями, распределенными по равномерному закону и равными 1 или -1.

Для низкочастотной части заполняются спектральные значения с индексом

$$i \in 1..50 \cup 206..255.$$

Для высокочастотной части соответственно $i \in 51..100 \cup 156..205$.

5. Расчет обратного преобразования Фурье от созданных массивов S_{re} , S_{im} .
6. Поиск максимального значения $Peak$ во временном представлении преамбулы, учитывая, что сигнал во временной области комплексный, т.е. для вычисления значения в каждой точке необходимо брать модуль от комплексного числа.

7. Если $GlobalPeak > Peak$, то выполняется перезапись переменной $GlobalPeak = Peak$ и значения компонент S_{re} , S_{im} сохраняются в результирующие массивы спектральных коэффициентов преамбулы $SPre_{re}$, $SPre_{im}$.
8. Если $Cnt < 100000000$, то переходим к шагу 4, иначе выводим сохраненные массивы $SPre_{re}$, $SPre_{im}$.

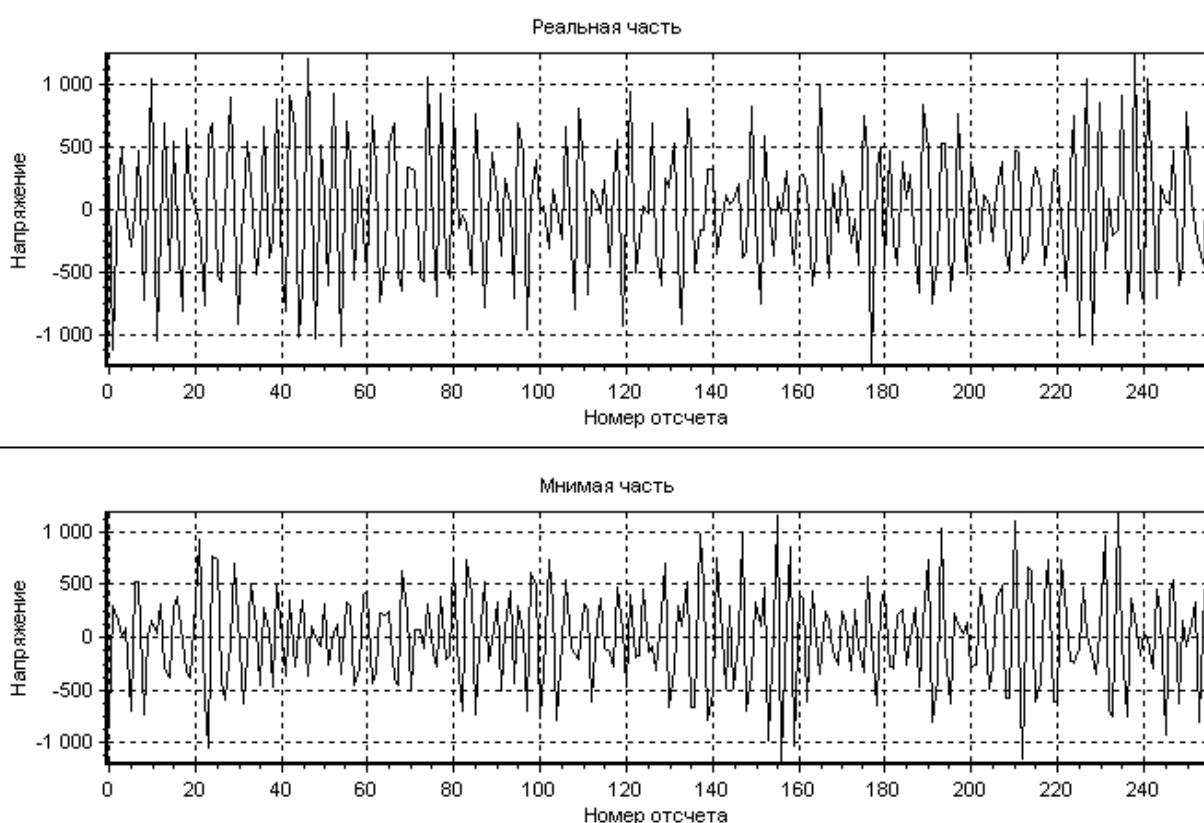


Рисунок 2.4.5 - Временное представление высокочастотной части преамбулы

При использовании представленного алгоритма за 100 миллионов итераций удалось найти коэффициенты преамбулы, которые для ВЧ части позволили получить пик-фактор 1,78 и для НЧ части 1,73. На стандартном компьютере с процессором *AMD64x2 2.4 ГГц*, для выполнения 1 миллиона итераций требуется порядка 10 секунд, т.о. 100 миллионов итераций выполняется менее чем за треть часа.

На рисунке 2.4.5 и рисунке 2.4.6 представлена оптимизированная преамбула *OFDM* без нормализации к диапазону АЦП для ВЧ и НЧ частей соответственно.

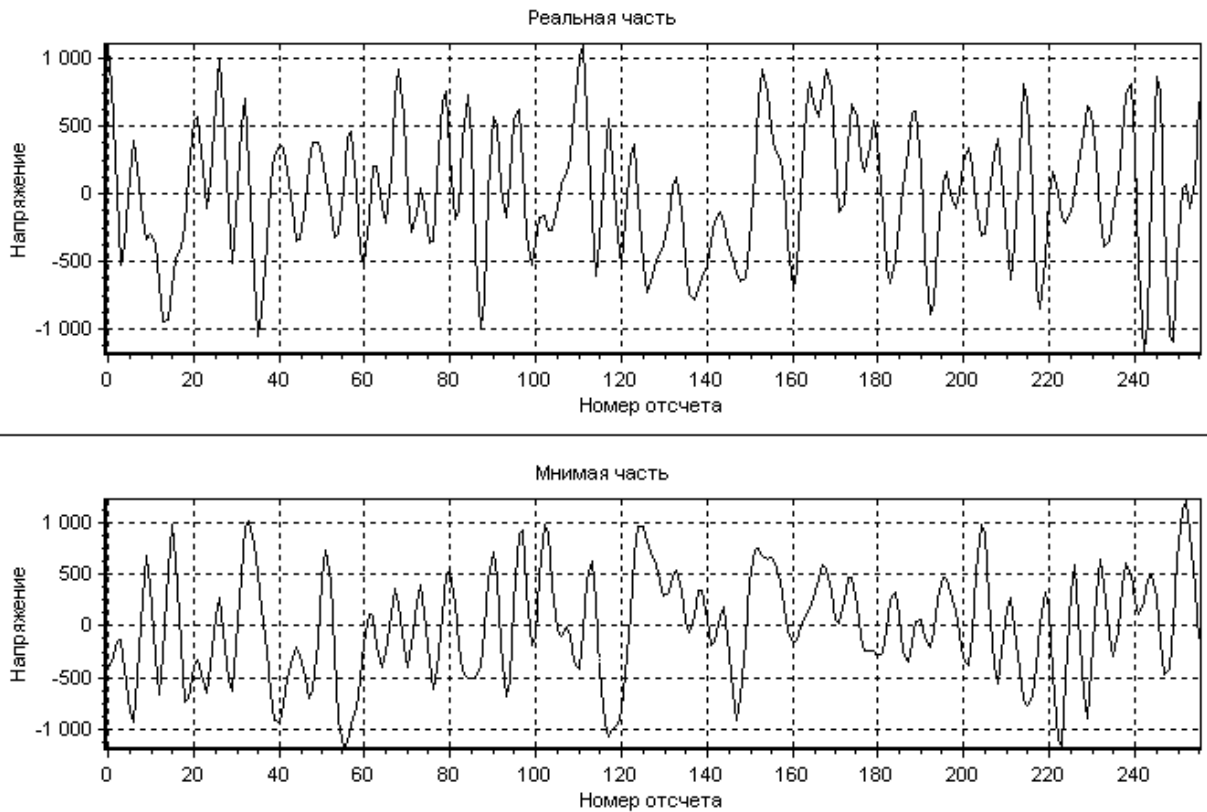


Рисунок 2.4.6 - Временное представление низкочастотной части преамбулы

Модифицированная преамбула по сравнению со стандартной позволила повысить ее энергию в 1,5 раза ($1,21^2$) или $\approx 1,7$ дБ. Однако необходимо отметить, что 1,7 дБ это будет выигрыш при расчете ОКПХ, а не при коррекции КПХ, т.к. погрешность получения символа данных будет значительно больше. Пусть погрешность расчета ОКПХ равна $\sigma_{\text{ОКПХ}}$, а погрешность получения данных σ_{DATA} , при этом общая погрешность

коррекции КПХ $\sigma_{\text{КПХ}} = \sqrt{\sigma_{\text{ОКПХ}}^2 + \sigma_{\text{DATA}}^2}$. Пусть $\sigma_{\text{DATA}} = \sigma$, тогда $\sigma_{\text{ОКПХ}} = 0,37 \sigma$,

т.к. уровень компонент преамбулы в 2,7 раза превосходят уровень данных, следовательно, погрешность расчета будет в тоже количество раз меньше.

Таким образом,

Если произвести те же самые расчеты для компонент преамбулы, используемой в стандартном алгоритме, то получим

$$s_{\text{КОР}} = \sqrt{\left(0,37 \cdot \left(\frac{2,16}{1,78}\right) \cdot s\right)^2 + s^2} \approx \sqrt{1,2} s.$$

Таким образом, выигрыш при коррекции в размах по напряжению составит $\sqrt{1,2} / \sqrt{1,13} \approx 1,03$ или $20\lg 1,03=0,26$ дБ.

Модифицированная преамбула была испытана на системе связи точка-многоточка *OFDM* со следующими параметрами сигнала:

1. Центральная частота 3495МГц.
2. Ширина полосы 7 МГц.
3. Длительность фрейма 70 символов.
4. Модуляции *QPSK* $1/2$, *QPSK* $3/4$, *QAM16* $1/2$, *QAM16* $3/4$, *QAM64* $2/3$, *QAM16* $3/4$.

Испытания показали уменьшение количества битовых ошибок на граничных условиях работы для всех модуляций.

Пик-фактор сигнала *OFDM* символа данных будет больше пик-фактора преамбулы, вследствие того, что в символ могут попасть любые данные и рандомизация не может обеспечить такое же минимальное пиковое значение. Стоит обратить внимание, что при одинаковом уровне максимальных значений спектральных компонент СКЗ сигналов *QPSK* > *QAM16* > *QAM64*, т.к. для модуляции *QPSK* все спектральные значения компонент максимальны, а для *QAM* сигналов максимальны уже не все компоненты и чем больше уровней в сигнале, тем меньше будет СКЗ. Т.о. для обеспечения одинакового пикового значения символа данных с преамбулой должны использоваться различные максимальные значения спектральных компонент в зависимости от модуляции. При разделении преамбулы на 2 символа и использовании уровня спектральных компонент преамбулы A_{pre} для модуляции *QPSK* должен быть выбран уровень $0,37A_{pre}$, для *QAM16* – $0,51A_{pre}$, *QAM64* – $0,525A_{pre}$. Т.о. для символа данных эффективное отношение сигнал/шум будет меньше для *QPSK* на $20\log(0,37)=8,6$ дБ, *QAM16* $20\log(0,51)=5,85$ дБ, *QAM64* $20\log(0,525)=5,59$ дБ.

2.5. Способ временной синхронизации для преамбулы, имеющей дискретно-непрерывный набор спектральных компонент

Для работы технологии *OFDM* необходимо, чтобы приемник синхронизировался по преамбуле и также по преамбуле рассчитал обратный комплексный коэффициент передачи - ОККП канала связи, который будет применяться для комплексного умножения на спектр каждого последующего символа [67].

Известны различные способы временной синхронизации, но все они работают с преамбулами, которые имеют повторяющиеся участки во временной области. Т.к. разработанный способ коррекции комплексной передаточной характеристики канала связи *OFDM* использует преамбулу, имеющую дискретно-непрерывный набор спектральных компонент, т.е. не имеющую повторяющихся участков во временной области, то известные способы синхронизации не могут быть использованы. В связи с чем возникла задача разработки нового способа временной синхронизации.

В системах связи, в том числе с подвижными объектами, каналы распространения сигнала между приемником и передатчиком данных являются многолучевыми и нестационарными. Эффективность систем связи во многом определяется способностью алгоритмов временной синхронизации обеспечить в многолучевых нестационарных каналах необходимую точность оценки временного положения сигнала.

Для временной синхронизации обычно используют специальный сигнал - преамбулу, которая предшествует информационному сообщению. Преамбула представляет собой символ или символы *OFDM* с заранее известным спектральным составом.

Разработанный алгоритм направлен на повышение помехоустойчивости временной синхронизации системы связи.

Поставленная задача решается за счет того, что решающая функция представляет собой квадрат модуля комплексной функции взаимной корреляции принятого сигнала и эталонной преамбулы без циклического

префикса, которая не передавалась через канал передачи, а была заранее известна. Преамбула представляет собой дельта - коррелированный шумоподобный сигнал. [68] Использование именно этого свойства при обработке приводит к повышению помехоустойчивости за счет резкого сужения области максимума решающей функции. Сужение области максимума легко увидеть, сравнивая рисунок 1.2.8 и аналогичные ему рисунки 2.5.1-2.5.2. На рисунке 1.2.8 показана временная метрика решающих функций без шумов и искажений для метода Шмидла и Кокса, метода А и метода В. Ширина решающей функции для метода В на уровне 0,1 составляет $\Delta_B \approx 200$ дискрет.

На рисунках 2.5.1 и 2.5.2. представлена нормированная решающая функция предлагаемого способа без шумов, начало преамбулы в точке с индексом равным $N_g/2$. Ширина решающей функции для предлагаемого способа на уровне 0,1 составляет $\Delta_c \approx 4$ дискрета, что приблизительно в 50 раз меньше чем у лучшего способа синхронизации из статьи [40].

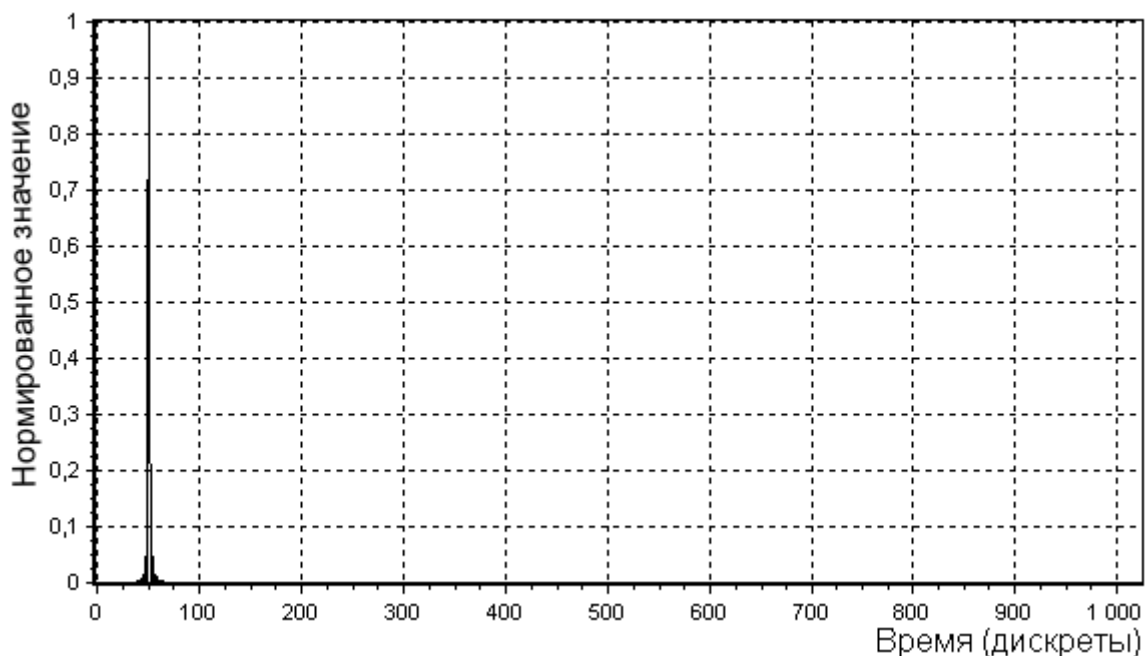


Рисунок 2.5.1. Нормированная решающая функция без шумов, начало преамбулы в точке с индексом по времени 51

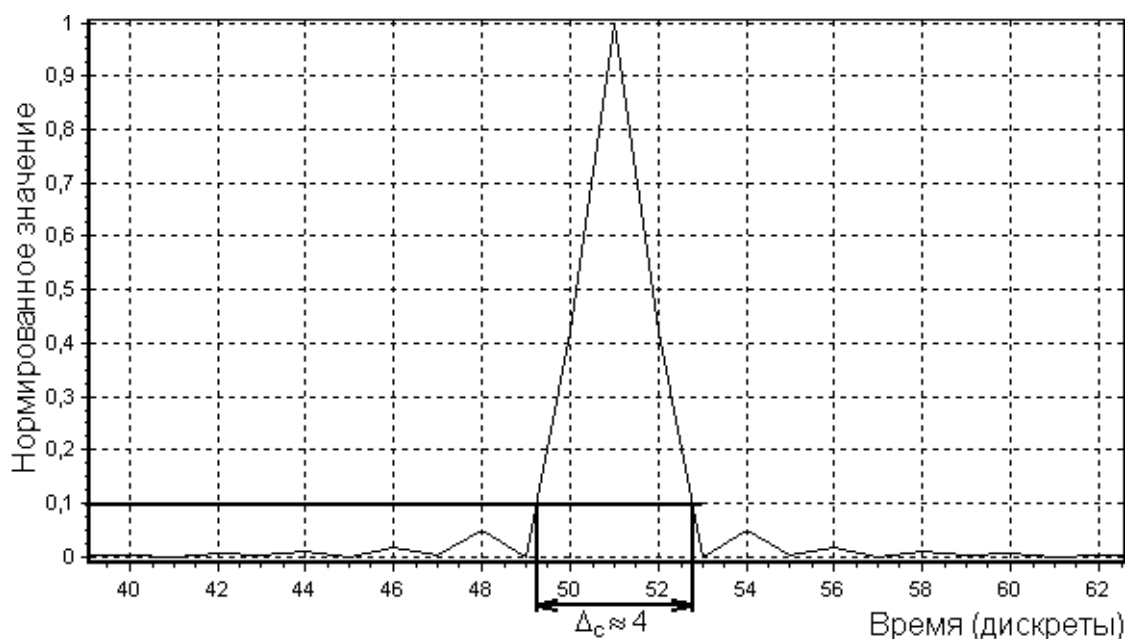


Рисунок 2.5.2. Нормированная решающая функция без шумов, начало преамбулы в точке с индексом по времени 51 (увеличенный масштаб по оси времени)

Необходимо отметить, что преамбула для предлагаемого способа может содержать один, два или более двух символов, каждый из которых во временной области не имеет повторяющихся участков. Это необходимо, чтобы максимум решающей функции был один для каждого символа, который используется для синхронизации.

Для проведения временной синхронизации на основе комплексной функции взаимной корреляции между принятым сигналом и эталонной преамбулой без циклического префикса, рассчитываемой через обратное преобразование Фурье от произведения спектра принятого сигнала на комплексно сопряженный спектр эталонной преамбулы, формируется решающая функция (2.5.1) [68], которая представлена на рисунках 2.5.1, 2.5.2.

$$M_K[d] = |IFFT[FFT(r[d]) \cdot P^*[f]]|^2 \quad (2.5.1)$$

где $IFFT$ – обратное преобразование Фурье;

FFT – прямое преобразование Фурье;

$r[d]$ –массив данных принятого сигнала, состоящий из N временных отсчетов;

N – количество точек в преобразовании Фурье;

$P_{\text{э}}^*[f]$ – массив комплексно сопряженных спектральных компонент эталонной преамбулы, т.е. преамбулы, которая не передавалась через канал передачи, а была заранее известна;

d – временной индекс для массивов $r[d]$ и $M_k[d]$.

В реальном канале связи время запаздывания зависит от частоты, т.е. имеется неравномерность группового времени запаздывания (ГВЗ), которая приводит к «растеканию» максимума решающей функции (см. нормированную решающую функцию для размерности преобразования Фурье $N = 1024$, длина циклического префикса $N_g = 256$ при прохождении сигнала преамбулы по реальному каналу связи на рисунках 2.5.3 – 2.5.4). Что приводит к затруднению при установке порогового коэффициента k для обнаружения преамбулы таким образом, чтобы не было ложных обнаружений и пропусков. Для устранения этого недостатка предлагается использовать сумму значений M_s решающей функции на участке значимости, т.е. в диапазоне индекса от a до b вместо значения максимума решающей функции (см. рисунок 2.5.4).

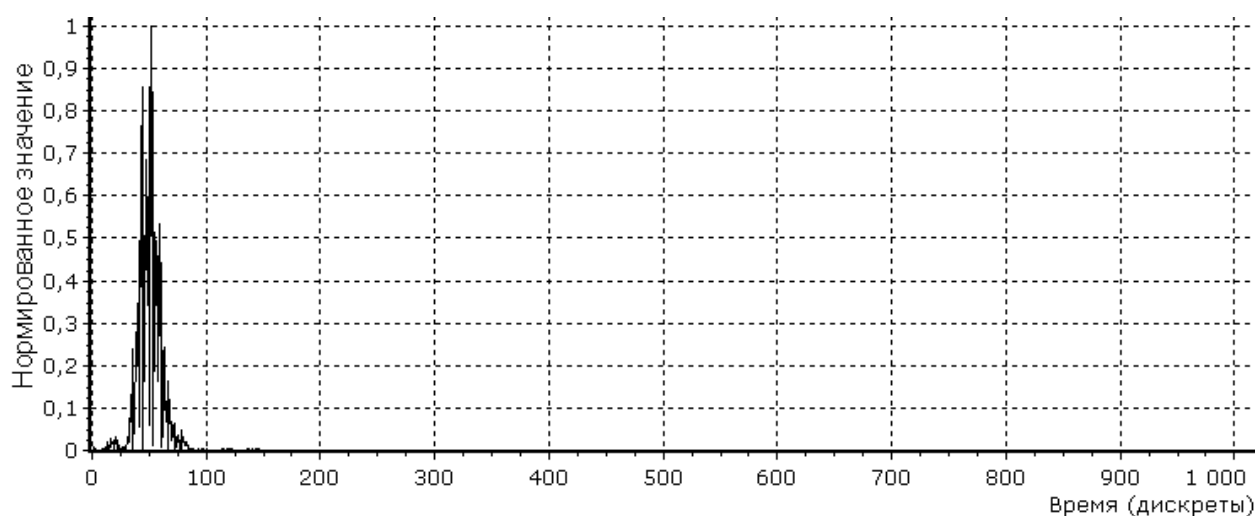


Рисунок 2.5.3 - Нормированная решающая функция при прохождении сигнала преамбулы по реальному каналу связи, размерность преобразования Фурье $N=1024$

Таким образом, основной критерий обнаружения преамбулы при растекании максимума решающей функции:

$$M_S > k \cdot A,$$

где k – порог обнаружения;

$$A = \frac{1}{N - (b - a + 1)} \left(\sum_{d=0}^{a-1} M_K[d] + \sum_{d=b+1}^{N-1} M_K[d] \right) - \text{среднее значение}$$

решающей функции, рассчитываемое в диапазоне индекса d от 0 до $N-1$, исключая из этого интервала диапазон индекса d от a до b .

Значение M_S рассчитывается на участке значимости как сумма значений решающей функции в диапазоне индекса d от a до b :

$$M_S = \sum_{d=a}^b M_K[d],$$

где a – это первый индекс, когда значение решающей функции больше отношения максимума решающей функции M к Q ;

b – последний индекс, когда значение решающей функции больше отношения максимума решающей функции M к Q .

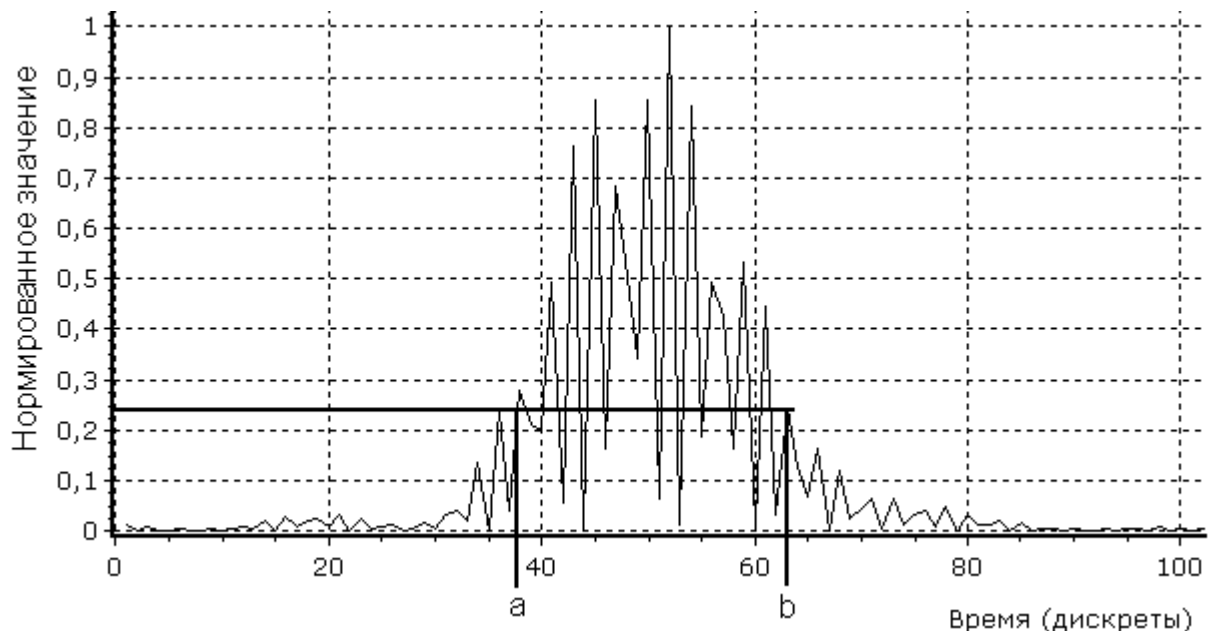


Рисунок 2.5.4 - Нормированная решающая функция при прохождении сигнала преамбулы по реальному каналу связи, размерность преобразования Фурье $N=1024$ (увеличенный масштаб)

Для снижения вероятности ложного обнаружения условие $M_S > k \cdot A$ целесообразно дополнить двумя условиями: $I < N - N/4$ и $b - a < N_g$,

где I - индекс, соответствующий временному положению максимума решающей функции $M_K[d]$, т.е. $M = M_K[I]$.

Условие $I < N - N/4$ не позволяет определять наличие преамбулы, если она найдена на отрезке решающей функции от $N - N/4$ до $N - 1$. Применимость этого условия обусловлена тем, что на следующем шаге подстройки (переменная $step = N/2$ в алгоритме обнаружения) произойдет обнаружение преамбулы в лучших условиях, т.к. большее количество дискрет преамбулы (большее на $step$ дискрет) попадет в массив данных принятого сигнала $r[d]$. Если $I \geq N - N/4$, то это означает, что в массив данных принятого сигнала $r[d]$ попало не более чем $N/4 + N_g$ дискрет (временных отсчетов) преамбулы.

Условие $b - a < N_g$ говорит о том, что «растекание» решающей функции на выбранном уровне значимости не может быть больше длины циклического префикса, другими словами ширина участка значимости должна быть меньше длины циклического префикса.

Значение порогового коэффициента k зависит от количества используемых компонент преамбулы и длительности циклического префикса N_g . Для исключения ошибок временной синхронизации необходимо, чтобы значение коэффициента k было больше, чем отношение M_S/A для дискретизированного сигнала преамбулы без шумов, при попадании в массив данных принятого сигнала только циклического префикса преамбулы, т.е. в массив $r[d]$ попадает N_g дискрет символа преамбулы. Такая установка порогового коэффициента гарантирует, что преамбула не будет определена при попадании в массив $r[d]$ только циклического префикса, если бы произошло такое определение, то временной индекс положения преамбулы I был бы ошибочно определен на отрезке от 0 до $N_g - 1$, реальный индекс максимума в этом случае позже на N дискрет, т.е. правильное временное положение преамбулы начинается в индексе времени $I + N$. При увеличении

длины циклического префикса и попадании в массив $r[d]$ только циклического префикса отношение M_g/A также увеличивается, поэтому значение коэффициента k должно быть соответственным образом увеличено.

Способ реализуется следующим образом. Принятый сигнал с выхода радиомодуля заданного уровня с нулевой промежуточной частотой оцифровывается АЦП, получается массив входных комплексных данных $r[i]$. Обработка данного массива осуществляется представленным ниже алгоритмом.

Алгоритм обнаружения преамбулы

Особенностью алгоритма является синхронизация в точку, соответствующую середине циклического префикса N_g . [69] При такой синхронизации обеспечивается наименьшее возможное наложение *OFDM* символов друг на друга, т.е. минимизируется межсимвольная интерференция.

1. Инициализация переменных: состояние $state=0$, сдвиг $shift=0$.
2. Рассчитывается решающая функция (2.5.1) в диапазоне от $shift$ до $shift+N-1$ дискрет. Решающая функция является квадратом модуля комплексной функции взаимной корреляции между принятым сигналом и заранее известной эталонной преамбулой без циклического префикса, которая определяется как обратное преобразование Фурье от комплексного произведения спектра принятого сигнала на комплексно сопряженный спектр эталонной преамбулы $P_g^*[f]$ (формула (2.5.1)). При этом спектр эталонной преамбулы известен заранее, а спектр принятой преамбулы получается после прямого преобразования Фурье от временной области принятого сигнала. Значение переменной $shift$ должно лежать в диапазоне возможных значений оцифрованного АЦП массива данных, т.е. если значение $shift+N-1$ больше максимального оцифрованного индекса в массиве $r[i]$, то переход к концу алгоритма с результатом: преамбула не обнаружена. Более подробно расчет решающей функции $M_k[d]$ описан в пп. 2.1-2.7.

- а. Определяется комплексный спектр принятого сигнала через прямое преобразование Фурье $r_f[f]=FFT(r[d])$,

где $f=0,1,2,..N-1$ – индекс частоты в спектре;

$d=0,1,2,...,N-1$ – индекс отсчета по времени в полученном массиве данных после сдвига на *shift* дискрет относительно начала;

N – количество точек в преобразовании Фурье.

- b. Устанавливается индекс частоты $f=0$.
- c. Если $f < N-1$, то переход к п. 2.4, иначе к 2.6.
- d. Определяется комплексная точка в массиве $S[f] = r_f[f] \cdot P_{\vartheta}^*[f]$.
- e. Инкрементирование индекса частоты $f=f+1$. Переход к п. 2.3.
- f. Определяется массив комплексной функций взаимной корреляции через обратное преобразование Фурье $corr[d] = \text{IFFT}(S[f])$.
- g. Рассчитывается массив решающей функции как квадрат модуля от массива $corr[d]$.

$$M_K[d] = corr_{Re}[d] \cdot corr_{Re}[d] + corr_{Im}[d] \cdot corr_{Im}[d],$$

где $d=0,1,2,...,N-1$;

$corr_{Re}[d]$, $corr_{Im}[d]$ действительные и мнимые части комплексного массива $corr[d]$ соответственно.

3. Определяется максимум M решающей функции $M_K[d]$ и индекс соответствующий данному максимуму – I , т.е. $M = M_K[I]$.

4. Определяется значение переменной M_Q : $M_Q = M/Q$.

5. Поиск индекса a , где a – это первый индекс, когда значение решающей функции больше отношения максимума M к Q . Поиск индекса осуществляется путем перебора индекса d от 0 до I : если $M_K[d] > M_Q$, тогда $a=d$.

6. Поиск индекса b , где b – это последний индекс, когда значение решающей функции больше отношения максимума M к Q . Поиск индекса осуществляется путем перебора индекса d от $N-1$ до I : если $M_K[d] > M_Q$, тогда $b=d$.

7. Рассчитывается сумма значений функции $M_K[d]$ на участке значимости, т.е. в интервале индекса d от a до b : $M_S = \sum_{d=a}^b M_K[d]$.

8. Рассчитывается среднее значение A решающей функции в интервалах от 0 до $a-1$ и от $b+1$ до $N-1$:

$$A = \frac{1}{N - (b - a + 1)} \left(\sum_{d=0}^{a-1} M_K[d] + \sum_{d=b+1}^{N-1} M_K[d] \right) - \text{среднее значение решающей}$$

функции, рассчитанное при исключении участка значимости.

9. Если $state=0$ (преамбула ни разу не обнаруживалась), то переход к шагу 10, иначе к шагу 13.

10. Если $M_S > k \cdot A$, $I < N - N/4$ и $b - a < N_g$ то переход к шагу 11, иначе к шагу 12, где k – порог обнаружения. Выполнение условий означает, что преамбула найдена, и необходимо установить сдвиг по времени таким образом, чтобы максимум функции был в середине циклического префикса (шаг 11). Иначе, сдвиг в плюс по времени на $step$ дискрет (шаг 12) и еще одна попытка обнаружить преамбулу первый раз.

11. $shift = shift + I - N_g/2$, $state=1$, переход к шагу 2. Настройка сдвига по времени таким образом, чтобы максимум функции был в середине циклического префикса.

12. $shift = shift + step$ - сдвиг времени на $step$ дискрет. Переход к шагу 2.

13. Если $M_S > k \cdot A$, $I < N - N/4$ и $b - a < N_g$, то переход к шагу 14, иначе к шагу 15.

14. $shift = shift + I - N_g/2$. Преамбула обнаружена. Переход к шагу 16.

15. Преамбула не обнаружена.

16. Конец.

Выбор порогового уровня M_Q в 25% ($Q=4$) относительно максимума позволяет легко реализовывать его расчет в аппаратуре за счет деления максимального значения M на 4 с использованием битового сдвига вправо на 2.

Для уменьшения количества итераций поиска преамбулы рекомендуется установить значение переменной $step=N/2$.

Для улучшения помехоустойчивости алгоритма значение переменной $step$ должно быть уменьшено, минимальное значение равно 1.

При реализации алгоритма в аппаратуре для непрерывного поиска преамбулы во время работы алгоритма входной массив $r[i]$ непрерывно заполняется новыми значениями, оцифрованными АЦП. В случае если необходимое количество точек для следующей итерации еще не оцифровано, то делается задержка выполнения алгоритма обнаружения преамбулы до момента времени, пока необходимое количество данных не будет собранно.

В результате выполнения алгоритма обнаружения преамбулы с результатом «преамбула обнаружена» идеальное значение индекса $I=N_g/2$, однако, из-за шумов и неравномерности ГВЗ может происходить допустимое отклонение от $N_g/2$, например, $\pm N_g/10$. Т.о. если $N_g/2+N_g/10 \geq I \geq N_g/2-N_g/10$, то синхронизация считается успешной и возможна дальнейшая обработка *OFDM* символов, иначе переходят в алгоритм обнаружения преамбулы.

Алгоритм поддержания временной синхронизации

Алгоритм основывается на начальной синхронизации через алгоритм обнаружения, постоянной длительности фрейма (как правило, составляющей несколько сотен символов) и небольшом расхождении между опорными частотами передающего и приемного модемов (как правило, не хуже 10^{-6}). Например, при длительности фрейма 400 символов и длительности одного символа 288 дискрет, имеем изменение индекса положения максимума на 0,1 дискрет ($400*288*10^{-6} \approx 0,1$ дискрет). Эти моменты обуславливают, что от фрейма к фрейму индекс положения максимума решающей функции без дополнительной подстройки меняется меньше, чем длительность циклического префикса (в условиях реального радиоканала меняется не более чем на несколько дискрет, резкие изменения обусловлены неравномерностью ГВЗ, при отсутствии резких изменений канала из-за несовпадения опорных частот возможно изменение на 1 дискрет). Т.о.

принимая преамбулу следующего фрейма, можно с уверенностью сказать, что максимум решающей функции лежит относительно предыдущего своего местоположения в диапазоне половины длительности циклического префикса. Все эти свойства упрощают алгоритм поддержания синхронизации. По шагам алгоритм может быть представлен:

1. $shift=0$.

Пункты 2-8 идентичны соответствующим пунктам алгоритма обнаружения преамбулы.

9. Если $M_S > k \cdot A$, $I < N_g$ и $b-a < N_g$, то преамбула обнаружена переход к пункту 10, иначе переход в алгоритм обнаружения преамбулы.

10. $\Delta = I - N_g/2$.

Для поддержания синхронизации текущий фрейм должен быть удлинен на Δ дискрет, за счет изменения защитного интервала, который находится в конце фрейма, это приведет к тому, что следующий фрейм будет принят позже на Δ дискрет и индекс положения максимума уменьшится на Δ дискрет.

Способ может использовать один или несколько символов преамбулы. Размерность преобразования Фурье (N) также может быть различной. В качестве примера была представлена временная область решающей функции для $N=1024$ (используется 1000 спектральных составляющих). При работе по двум символам, представленный выше алгоритм выполняется дважды, при этом дополнительно возникает возможность усреднить результаты и сделать выводы о неравномерности ГВЗ по значениям индексов a и b в зависимости от номера символа преамбулы.

Способ может быть реализован в устройстве *DMT* модема (англ. *Discrete multitone modulation* – дискретная многотональная модуляция) или *OFDM* модема, типовая блок-схема которого представлена на рисунке 2.5.5, а также в любом другом устройстве, которое позволит реализовать представленный выше алгоритм работы. Модули расчета прямого и обратного преобразований Фурье обязательно присутствуют во всех *OFDM* и

DMT модемах, что упрощает аппаратную реализацию предлагаемого способа.

Способ был апробирован в реальных условиях радиоканала на основе *OFDM* модема (см. рисунок 2.5.5). На рисунке 2.5.5 обозначены: микроконтроллер 1, радиомодуль 2, программируемое постоянное запоминающее устройство (ППЗУ) 3, оперативное запоминающее устройство (ОЗУ) 4, программируемая логическая интегральная схема (ПЛИС) 5, цифро-аналоговый преобразователь (ЦАП) 6, аналого-цифровой преобразователь (АЦП) 7.

Опорная частота $f_{\text{опорная}}$ преобразуется в ПЛИС в модуле преобразования частоты на основе ФАПЧ в тактовую частоту работы ПЛИС, частоту дискретизации ЦАП и частоту дискретизации АЦП. Опорная частота изменяется в небольших пределах за счет управляющего воздействия с микроконтроллера. Микроконтроллер взаимодействует с ПЛИС и интерфейсами ввода-вывода (интерфейс USB, сетевой интерфейс, последовательный интерфейс) через прямой доступ в память и соответствующие прерывания. Микроконтроллер подключен к энергонезависимой памяти ППЗУ и оперативной памяти ОЗУ через интерфейс памяти, что позволяет создавать ресурсоемкие алгоритмы обработки и управления. В задачи микроконтроллера входит управление режимами работы ПЛИС и радиомодуля, а также высокоуровневая обработка данных. В ПЛИС реализуются критические по скорости процедуры, такие как прямое и обратное преобразования Фурье, помехоустойчивое кодирование, шифрование.

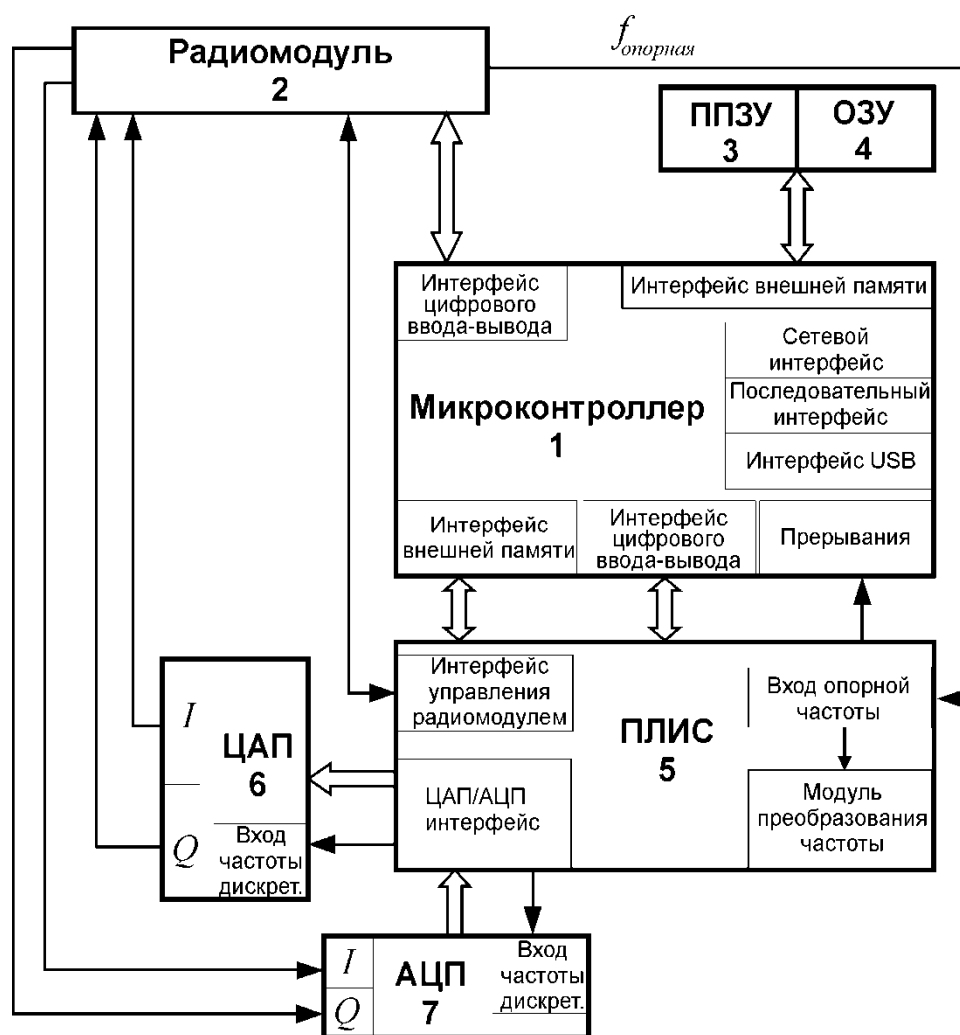


Рисунок 2.5.5 - Блок-схема цифрового модема

Кратко алгоритм работы передатчика цифрового модема можно представить так: передаваемая информация поступает на порт сетевого интерфейса, данные шифруются, рандомизируются, кодируются (вносится избыточность), перемешиваются, модулируются, с использованием обратного преобразования Фурье создаются *OFDM* символы, которые подаются на вход ЦАП, выход которого соединен с входом радиомодуля. ЦАП генерирует *OFDM* сигнал с нулевой промежуточной частотой, поэтому задача радиомодуля перенести спектр сигнала с выхода ЦАП на заданную несущую частоту.

Краткий алгоритм работы приемника цифрового модема можно описать следующим образом. Выходной сигнал заданного уровня с нулевой промежуточной частотой с выхода радиомодуля оцифровывается АЦП. По преамбуле, оцифрованной АЦП производится временная синхронизация и

рассчитывается массив $K_{inverse}[if_s/N]=U_{PRE}[if_s/N]/U_{RCV}[if_s/N]$ значений обратного комплексного коэффициента передачи,

где f_s – частота дискретизации;

N – количество точек в массиве данных преобразования Фурье;

U_{PRE} –массив заранее известных спектральных составляющих эталонной преамбулы;

U_{RCV} –массив спектральных составляющих преамбулы, прошедших через канал передачи;

i – целое число от 1 до $2z$ и от $N-2z$ до $N-1$;

$4z$ - количество используемых компонент.

Все последующие спектры *OFDM* символов данных умножаются на соответствующий коэффициент из массива $K_{inverse}[if_s/N]$, далее данные демодулируются, деперемешиваются, декодируются, дерандомизируются и дешифрируются и передаются на выход через сетевой интерфейс.

Наибольший эффект от использования предложенного способа временной синхронизации может быть достигнут в цифровых средствах связи реализованных по технологии *OFDM* и *DMT*. Предлагаемый способ существенно повышает помехоустойчивость временной синхронизации системы связи и расширяет область ее применимости.

На разработанный способ временной синхронизации подана заявка на патент РФ на Способ временной синхронизации системы связи на основе ортогонального частотного разделения каналов с мультиплексированием по преамбуле (заявка от 17.03.15 № 2015109441) и получено положительное решение о выдаче патента РФ от 25.10.2016.

2.6. Тактовая и частотная синхронизация в канале связи *OFDM*

Под тактовой синхронизацией понимается синхронизация частот дискретизации.

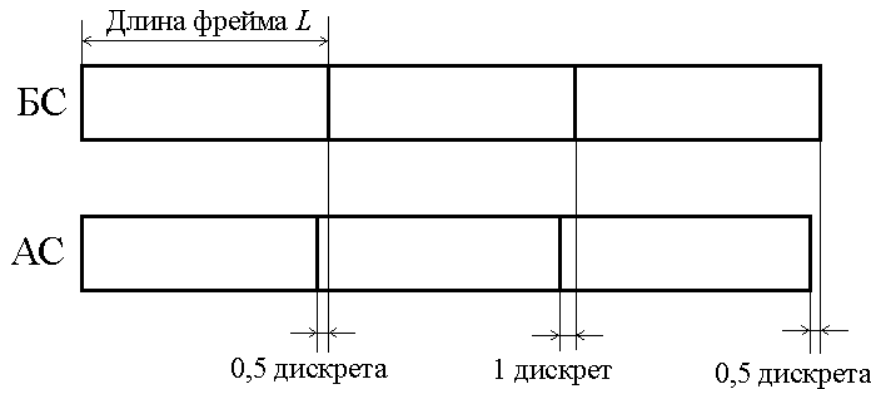


Рисунок 2.6.1 - Тактовая синхронизация

На рисунке 2.6.1 обозначено: БС – базовая станция, АС – абонентская станция. Длительность фрейма $L = 100$ OFDM символов.

Для определения ошибки частоты необходимо найти относительную ошибку в дискретах Δ на длительность фрейма L (см. рисунок 2.6.1):

$$\Delta = L_{BC} - L_{AC}.$$

Коррекция осуществляется только тогда, когда значение $|\Delta| \geq 1$ дискрет.

$$\frac{N}{f_s} = \frac{N + \Delta}{f_x};$$

$$N \cdot f_x = f_s \cdot (N + \Delta);$$

$$f_x = \frac{f_s(N + \Delta)}{N} = f_s + \frac{f_s \cdot \Delta}{N};$$

Отсюда ошибка частоты:

$$f_s - f_x = \frac{\Delta}{N} \cdot f_s.$$

Где f_s – частота дискретизации БС (эталонная частота), f_x – частота дискретизации АС, N – расстояние в дискретах между коррекциями (на рисунке 2.6.1 это расстояние равно $2L$).

Данные расчеты реализованы в алгоритме тактовой синхронизации. Особенностью алгоритма является синхронизация в точку, соответствующую середине циклического префикса.

Алгоритм:

1. Инициализируются переменные $CP_OFFSET_SS=16$ (для циклического префикса равного 32), $Frame_Number=0$, $First_Correction=0$.
2. Ожидание данных по новому принятому фрейму. $Frame_Number=Frame_Number+1$.
3. Рассчитывается $Correction = Corr_Offset - CP_OFFSET_SS$ (за счет вычитания будет синхронизации в середину циклического префикса).
 $Corr_Offset$ – найденное значение индекса максимума коэффициента корреляции.
4. Если $Correction$ не равно 0, тогда переход к п. 5, иначе к п. 8.
5. Изменение длительности фрейма на значение $Correction$.
6. Записывается номер фрейма, когда произошла последняя коррекция:
 $Last_Correction = Frame_Number$.
 $Frame_Number$ – номер фрейма, когда выполнялась коррекция.
7. Рассчитывается сумма коррекций $Correction_Sum = Correction_Sum + correction$.
8. Если $Frame_Number=1000$, переход к п. 9, иначе к п. 2.
9. Рассчитывается ошибка по частоте:

$$Freq_Error = \frac{Correction_Sum}{(Last_Correction - First_Correction) \cdot Frame_Length} \cdot fs$$

fs – частота дискретизации, $Frame_Length$ - длина фрейма в дискретах.

В соответствии со значением $Freq_Error$ производится коррекция управляющего напряжения (изменяется код, подаваемый на низкочастотный ЦАП), подающегося на генератор опорной частоты.

10. $First_Correction=Last_Correction$, $Correction_Sum=0$. Переход к п. 2.

Следует отметить, что если генератор опорной частоты является единым для частоты дискретизации и несущей частоты радиосигнала, то подстраивая по данному алгоритму генератор опорной частоты, чтобы выровнять частоту дискретизации абонентской станции по базовой станции

также будет происходить частотная синхронизация по несущим частотам. Эта особенность позволяет одновременно с тактовой синхронизацией получить и частотную.

Выводы по главе 2

1. Показано, что стандартный алгоритм коррекции КПХ требует перевода данных в полярную систему координат.
2. Предложен модифицированный алгоритм коррекции КПХ, позволяющий работать без использования полярной системы координат за счет преамбулы содержащей все частотные компоненты.
3. Представлен целочисленный алгоритм коррекции КПХ системы связи *OFDM*, который не использует в расчетах полярную систему координат и может быть легко реализован в аппаратуре связи.
4. Показано, что использование стандартной преамбулы на всех ее частотных компонентах приводит к значительному росту пик-фактора, поэтому для модифицированного алгоритма расчета КПХ требуется модифицировать преамбулу для минимизации пик-фактора.
5. Показано, что алгоритм генерации преамбулы для минимизации пик-фактора может быть реализован на основе создания преамбулы через генератор случайных чисел.
6. Показано, что при временной синхронизации применение в качестве решающей функции квадрата модуля функции взаимной корреляции и преамбулы имеющей дискретно-непрерывный набор спектральных компонент приводит к повышению помехоустойчивости за счет резкого сужения области максимума решающей функции.
7. Рассмотрено применение временной синхронизации для реализации тактовой.
8. Показано, что в случае единого опорного генератора выполнение тактовой синхронизации приведет к частотной синхронизации.

Глава 3. Анализ возможностей разработанных алгоритмов методами математического моделирования и апробации

В параграфе 3.1 приведен алгоритм генерации аддитивного Гауссового шума в заданной полосе частот, который необходим для моделирования алгоритмов рассмотренных во второй главе.

В параграфе 3.2 рассмотрена разработанная программа моделирования коррекции КПХ системы связи.

В параграфе 3.3 приведены результаты моделирования стандартного алгоритма WiMAX 802.16.2004 для коррекции КПХ системы связи, использующего в расчетах полярную систему координат.

В параграфе 3.4 приведены результаты моделирования модифицированного алгоритма коррекции КПХ системы связи, использующего в расчетах ортогональную систему координат.

В параграфе 3.5 приведено сравнение алгоритмов коррекции КПХ, работающих с вещественными значениями, по преамбуле и пилот-поднесущим.

В параграфе 3.6 проведена оценка точности предложенного целочисленного алгоритма коррекции КПХ по сравнению с вещественным алгоритмом.

В параграфе 3.7 представлены результаты моделирования предложенного способа временной синхронизации для преамбулы, имеющей дискретно-непрерывный набор спектральных компонент.

3.1. Алгоритм генерации шума в заданной полосе частот

Шум создается в соответствии со следующим алгоритмом: [70]

1. В соответствии с нормальным законом распределения задается функция плотности вероятности (закон распределения Гаусса):

$$f(x) = \frac{1}{\sqrt{2\pi} \sigma} e^{\frac{-x^2}{2\sigma^2}};$$

где σ – параметр масштаба распределения. [71]

2. Пересчитывается значение s в соответствии с формулой:

$$s = s \sqrt{\frac{f_o}{2(f_B - f_H)}}, \text{ которая позволяет учесть уменьшение полосы частот,}$$

занимаемой шумом. [72] Где f_o - частота дискретизации, f_H - нижняя граница полосы пропускания шума, f_B - верхняя граница полосы пропускания шума.

Значение s увеличивается в корень из уменьшения полосы шума раз. Это необходимо, чтобы скомпенсировать уменьшение s при фильтрации, т.о. после фильтрации s будет приблизительно равна заданному изначально значению.

3. Устанавливается интервал изменения переменной x [x_{min} , x_{max}]. Где $x_{min} = -3s$, а $x_{max} = 3s$ (закон распределения Гаусса). Вероятность того, что случайная величина находится в этих пределах равна 99,73%. Для каждой точки из интервала с шагом $\Delta = (x_{max} - x_{min})/k$ (где k – количество интервалов) определяется значение функции плотности вероятности и рассчитывается вероятность $P(x)$ в интервале $[x, x+step)$, $P(x) = f(x) * \Delta$. Рассчитывается количество точек для интервала $N_x = P(x) * N$, где N – общее количество точек. Создается результирующий линейный массив шумовых значений, в котором значения шума заданного уровня x повторяются N_x раз. Пример линейного массива показан в таблице 3.1.1.

Таблица 3.1.1. Массив данных шума.

№	0	1	2	3	4	...	994	995	996	997	998	999
Значение	x_{min}	x_{min}	x_1	x_1	x_1	...	x_{k-2}	x_{k-1}	x_{k-1}	x_{k-1}	x_{max}	x_{max}

Таблица 3.1.2. Пример результирующего массива шума.

№	0	1	2	3	4	...	994	995	996	997	998	999
Значение	x_3	x_{39}	x_{396}	x_{329}	x_{153}	...	x_{155}	x_{955}	x_{539}	x_{53}	x_{58}	x_{99}

4. Случайным образом выбирается индекс (выбор индекса осуществляется по равномерному случайному закону, который легко доступен во всех современных языках программирования, в диапазоне от 0 до $N-1$) точки из линейного массива данных шума, и если он (индекс) еще не использовался, то значение шума в этой точке копируется в результирующий

массив шума, иначе операция случайного выбора индекса повторяется до тех пор, пока не будет найден неиспользованный индекс. Устанавливается, что данный индекс был использован. Пример результирующего массива шума показан в таблице 3.1.2.

5. Умножается полученный массив шума на временное окно (например, окно Хэмминга).

6. Берётся прямое преобразование Фурье, т.о. получается спектр шума $S_{ш_w}$.

7. Фильтруется спектр $S_{ш_w}$, остаются только те спектральные составляющие, для которых

$$i \in \left[\text{round}\left(\frac{N \cdot f_H}{f_\partial}\right), \text{round}\left(\frac{N \cdot f_B}{f_\partial}\right) \right] \cup \left[\text{round}\left(\frac{N \cdot (f_\partial - f_B)}{f_\partial}\right), \text{round}\left(\frac{N \cdot (f_\partial - f_H)}{f_\partial}\right) \right].$$

Получается отфильтрованный спектр шума $S_{ш_w_ф}$.

8. Находится обратное преобразование Фурье от отфильтрованного спектра, получается массив шума во временной области $x_{ш_w_ф}$.

9. Для восстановления амплитуды – получения массива $x_{ш_ф}$, каждый элемент массива $x_{ш_w_ф}$ делится на функцию (массив коэффициентов) временного окна.

Из-за краевых эффектов после обратного преобразования Фурье 25% от начала и конца массива $x_{ш_ф}$ целесообразно отбросить.

10. Сигнал, передаваемый в ЦАП для генерации, будет составлять алгебраическую сумму полезного сигнала и отфильтрованного в заданной полосе шума $x + x_{ш_ф}$.

Параметр σ задается исходя из формулы: $g = 20 \lg(A/\sigma)$,

где A – СКЗ сигнала, g - отношение сигнал/шум в децибелах.

Откуда выражается σ : $s = \frac{A}{10^{g/20}}$.

3.2. Программа моделирования коррекции КПХ

На основе представленных в П.1 алгоритмов для моделирования коррекции КПХ была разработана программа на *Delphi* (см. рисунки 3.2.1-3.2.3), которая позволила произвести первичную отладку предложенных алгоритмов и убедиться в их работоспособности. [73, 74]

Программа предусматривает возможность выбора цифровых видов модуляции - *BPSK*, *QPSK*, *QAM16*, *QAM64*, изменения размера циклического префикса (4, 8, 16, 32, 64 дискрета), возможность разбиения преамбулы на несколько символов (2, 4, 8, 16 символов), возможность задавать сдвиг от начала символа (параметр, который определяет момент от начала символа, когда начинается запись принятых данных в массивы), для *OFDM* это значение не может превышать величину циклического префикса. Имеется возможность изменять значение частоты дискретизации и использовать окно Хэмминга при построении импульсной характеристики. [75]

Программа позволяет генерировать последовательность цифровых данных для разных типов ЦАП: 8/10/12/14/16 битных.

Программа использует нормальный закона распределения шума, позволяет вводить различные отношения сигнал/шум, дает возможность использовать окно Хэмминга при построении шума, а также предусматривает фильтрацию шума (при необходимости можно указать полосу пропускания шума). Необходимо отметить, что фильтрация нужна для того, чтобы более корректно провести моделирование, т.е. задать шумы только на используемых частотах: в интервалах 0 - 1563 кГц и 2437 - 4000кГц т.е. во всем рабочем диапазоне из 200 активных спектральных компонент.

Программа позволяет выводить на графики следующие данные: гистограмму распределения шума, гистограмму распределений максимального уровня принятого сигнала, ОКПХ, принятый сигнал после эквалайзера (частотная область), принятый сигнал (частотная область), переданный сигнал с шумом (временная область). А также определяет значение доверительного интервала для вероятности 90% , значение СКО и

среднее значение амплитуды. Для расчета этих параметров, а также для построения гистограммы распределений максимального уровня принятого сигнала используется кнопка «1000 изм», позволяющая произвести расчет для 1000 измерений.

Для создания *OFDM* символа используются случайные данные. Кнопка «Заполнить» позволяет получить случайные данные для формирования *OFDM* символа. Необходимо учитывать, что для формирования 1 *OFDM* символа необходимо минимум 24 байт, если модуляция *BPSK*, 48 байт, если *QPSK*, 96 байт, если *QAM16*, 144 байт, если *QAM64*.

Для корректной работы программы необходимо задавать все необходимые параметры для генерации сигнала и шума в соответствующие поля ввода.

Список возможностей программы моделирования:

1. Генерация *OFDM* сигналов с цифровой модуляцией: *BPSK*, *QPSK*, *QAM16*, *QAM64*.
2. Генерация заданного количества символов преамбулы.
3. Генерация шума по закону распределения Гаусса.
4. Генерация шума в заданной полосе частот.
5. Генерация данных для ЦАП различных битностей.
6. Возможность задавать соотношение сигнал/шум.
7. Возможность изменять размер циклического префикса.
8. Возможность устанавливать частоту дискретизации.
9. Возможность получать информацию о СКО уровня поднесущих принятого сигнала от идеального, доверительном интервале для вероятности 90% и средней амплитуде спектральных компонент.
10. Детально работать с графиками: увеличивать каждый график, приближать отдельные элементы.
11. Возможность выводить на графики данные в кГц.

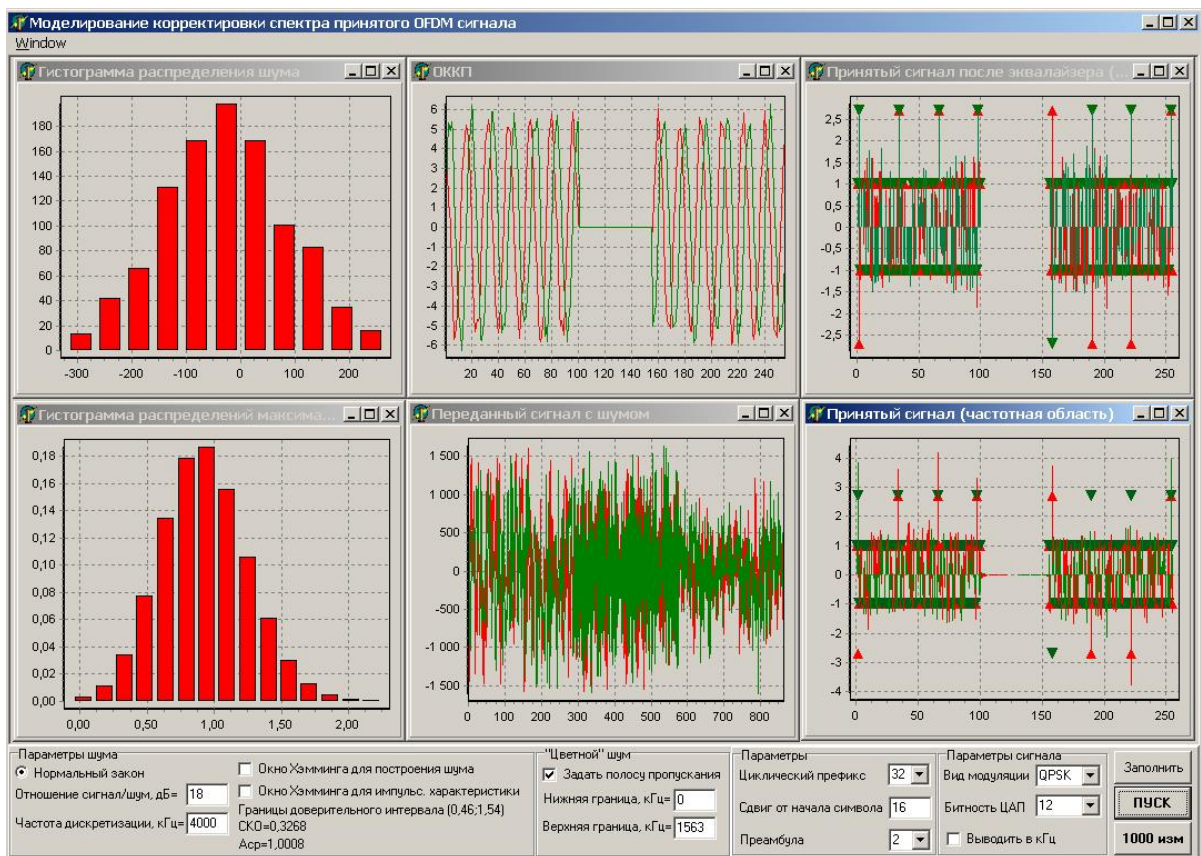


Рисунок 3.2.1 - Рабочее окно программы моделирования



Рисунок 3.2.2 - Рабочее окно программы моделирования, принятый сигнал после эквалайзера

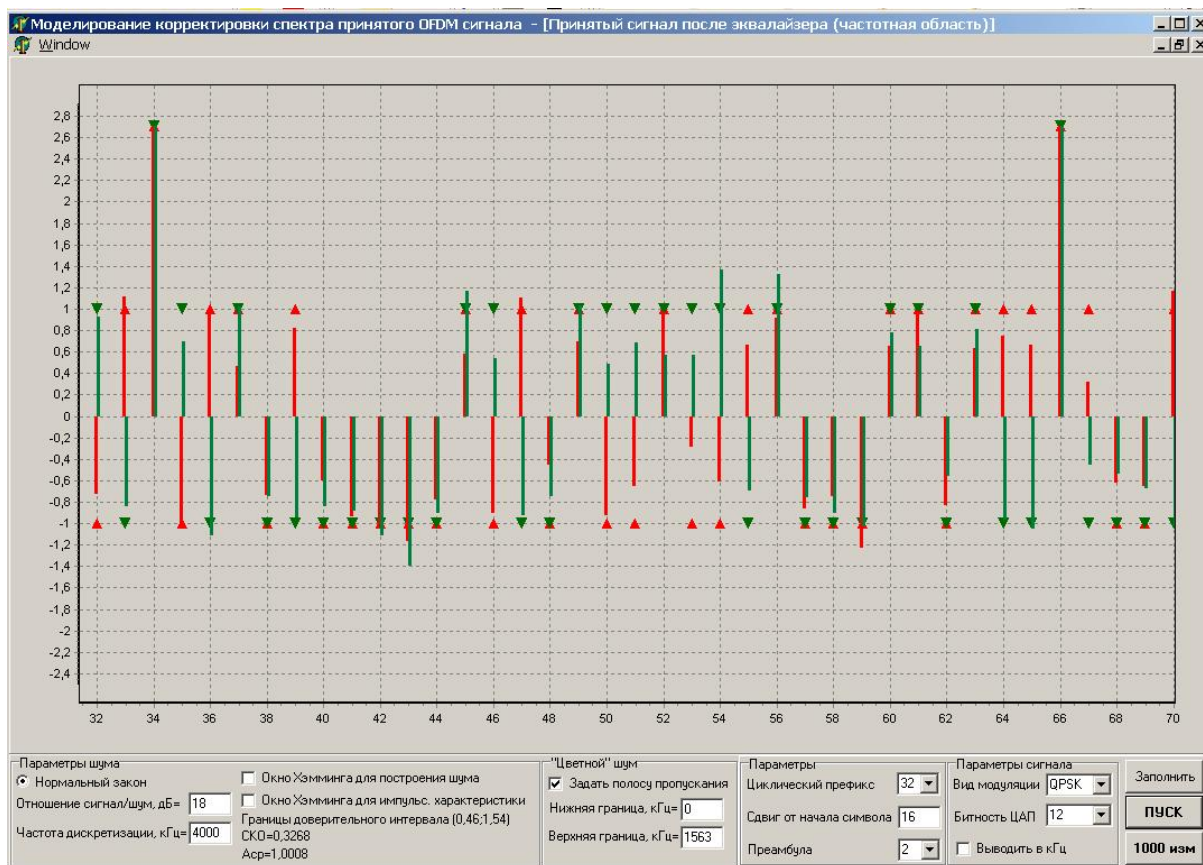


Рисунок 3.2.3 - Часть принятого сигнала после эквалайзера

3.3. Моделирование алгоритма коррекции КИХ, использующего в расчетах полярную систему координат

Для моделирования использовалась модуляция *QPSK*. Т.к. именно эта модуляция позволяет наиболее просто реализовать моделирование. В спектре *QPSK*-модулированного сигнала содержатся компоненты одного уровня, поэтому можно обойтись построением одной гистограммы. А, например, *QAM64*-модулированный сигнал содержит компоненты четырех различных уровней (1, 3, 5, 7), поэтому для реализации моделирования на примере этой модуляции необходимо было бы построить несколько гистограмм (для каждого уровня отдельная гистограмма) или обойтись одной, которая содержала бы информацию об отклонении от идеального значения.

На рисунках 3.3.1 - 3.3.6 представлены гистограммы распределений максимального уровня принятого сигнала, где b – битность ЦАП, SNR – отношение сигнал/шум, σ – СКО, d – доверительный интервал.

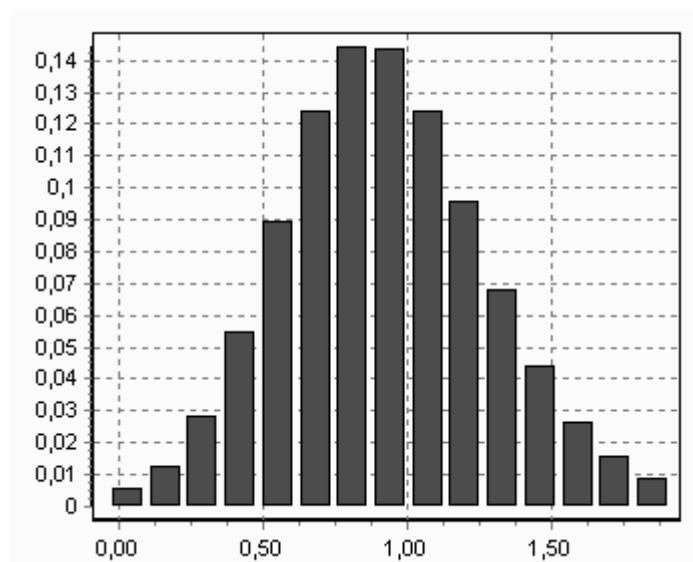


Рисунок 3.3.1 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=18$ дБ, $\sigma=0,3822$, $d(0,37;1,63)$)

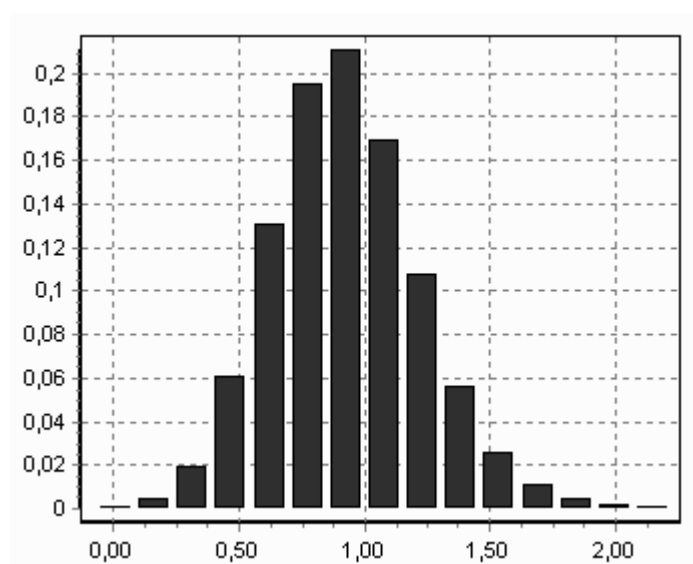


Рисунок 3.3.2 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=20$ дБ, $\sigma=0,3003$, $d(0,51;1,49)$)

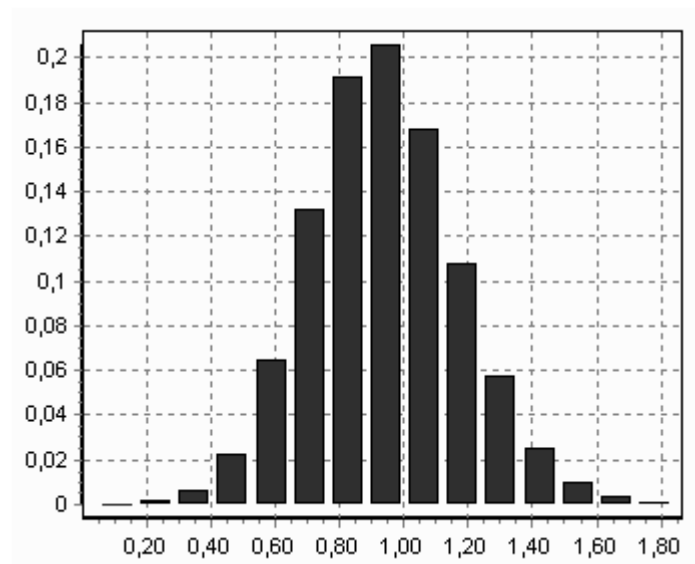


Рисунок 3.3.3 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=22$ дБ, $\sigma=0,2343$, $d(0,61;1,39)$)

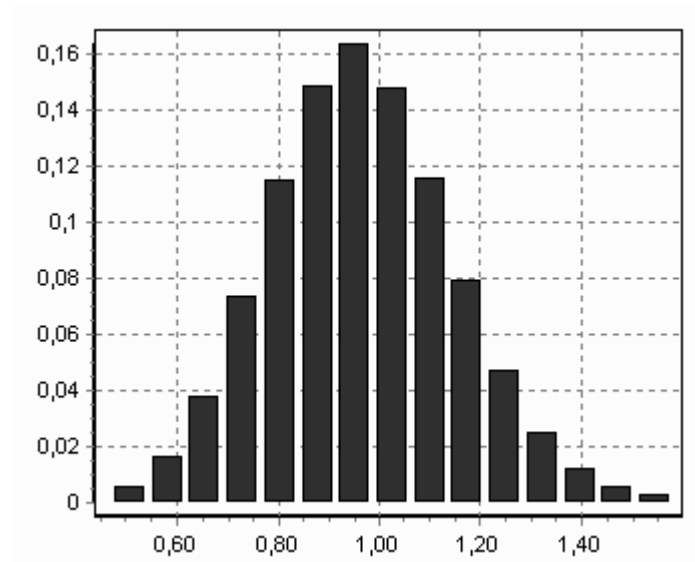


Рисунок 3.3.4 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=24$ дБ, $\sigma=0,1868$, $d(0,69;1,31)$)

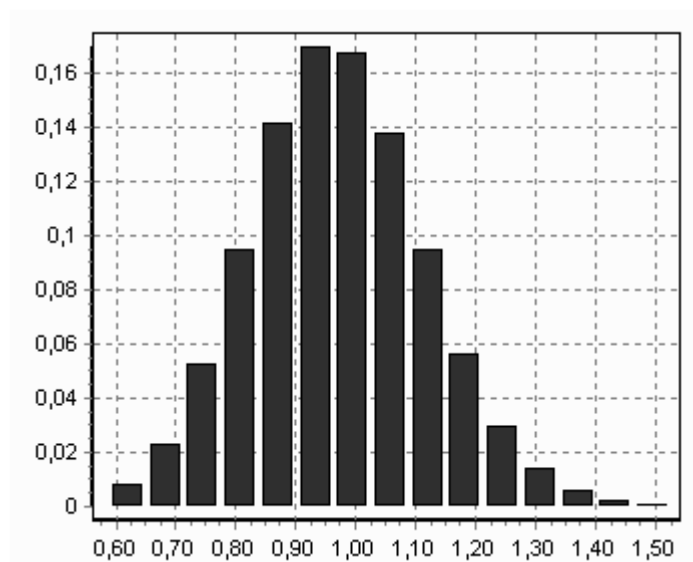


Рисунок 3.3.5 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=26$ дБ, $\sigma=0,1468$, $d(0,76;1,24)$)

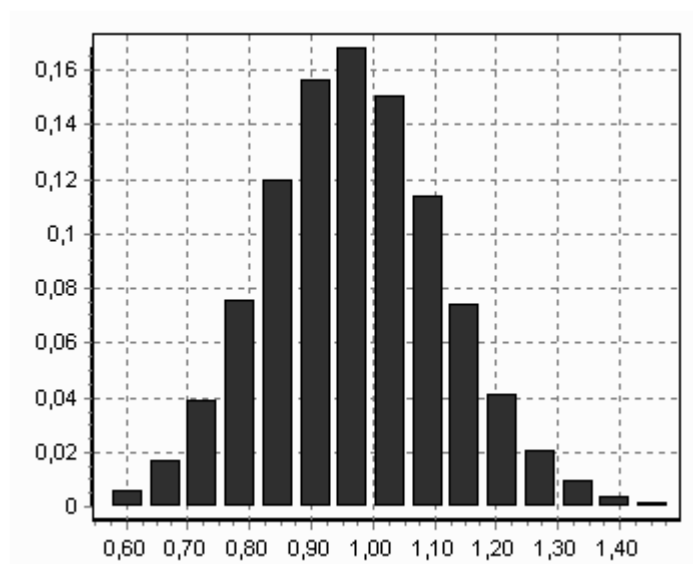


Рисунок 3.3.6 - Гистограмма распределений максимального уровня принятого сигнала ($b=16$ бит, $SNR=26$ дБ, $\sigma=0,1463$, $d(0,76;1,24)$)

С увеличением отношения сигнал/шум значение σ уменьшается. Найденные значения СКО будут использованы при сравнении алгоритмов.

В П.3 представлены гистограммы моделирования алгоритма коррекции КПХ, использующего в расчетах полярную систему координат с увеличенными по уровню значениями пилот-поднесущих. Уровень пилот-поднесущих увеличен до того же значения, что и в модифицированном алгоритме, который использует в расчетах ортогональную систему координат и составляет 2,7 (8,6 дБ) по отношению к информационным

поднесущим. Уровень пилот-поднесущих по стандарту WiMAX 802.16.2004 составляет 1,3 (2,5 дБ) по отношению к информационным поднесущим.

3.4. Моделирование предложенного алгоритма коррекции КПХ

В П.2 представлены гистограммы моделирования модифицированного алгоритма коррекции с уменьшенным значением СКЗ преамбулы, что позволило сделать пиковое значение преамбулы таким же, как и в случае стандартной преамбулы, в которой частотные компоненты идут через одну составляющую.

В П.4 представлены гистограммы распределений максимального уровня принятого сигнала при различных значениях параметра «сдвиг от начала символа», который определяет момент от начала символа, когда начинается запись принятых данных в массивы.

На рисунках 3.4.1-3.4.7 представлены гистограммы распределений максимального уровня принятого сигнала, где b – битность ЦАП, SNR – отношение сигнал/шум, σ – СКО, d – доверительный интервал.

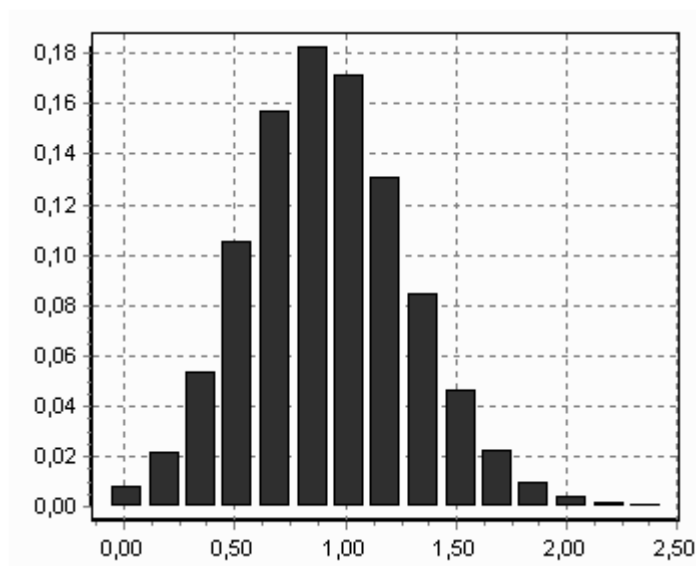


Рисунок 3.4.1 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=17$ дБ, $\sigma=0,3682$, $d(0,39;1,61)$)

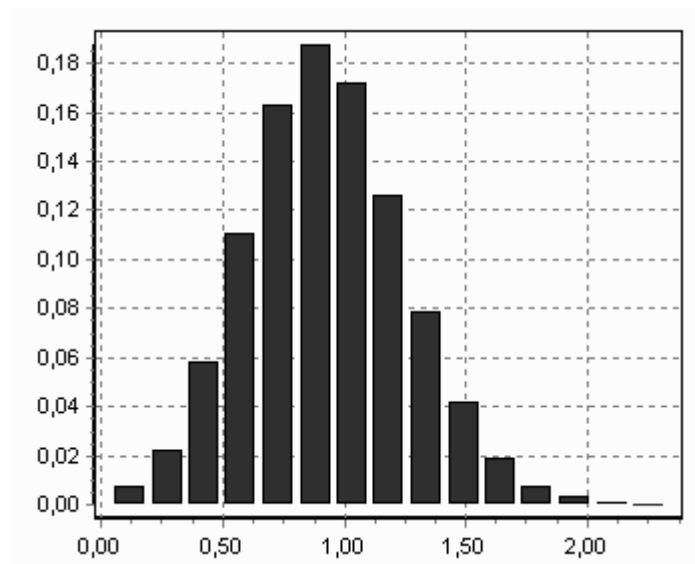


Рисунок 3.4.2 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=18$ дБ, $\sigma=0,3268$, $d(0,46;1,54)$)

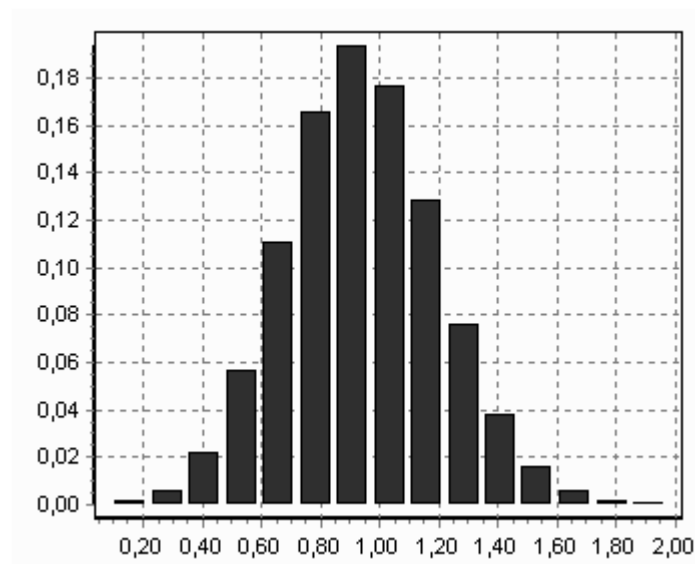


Рисунок 3.4.3 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=20$ дБ, $\sigma=0,2591$, $d(0,57;1,43)$)

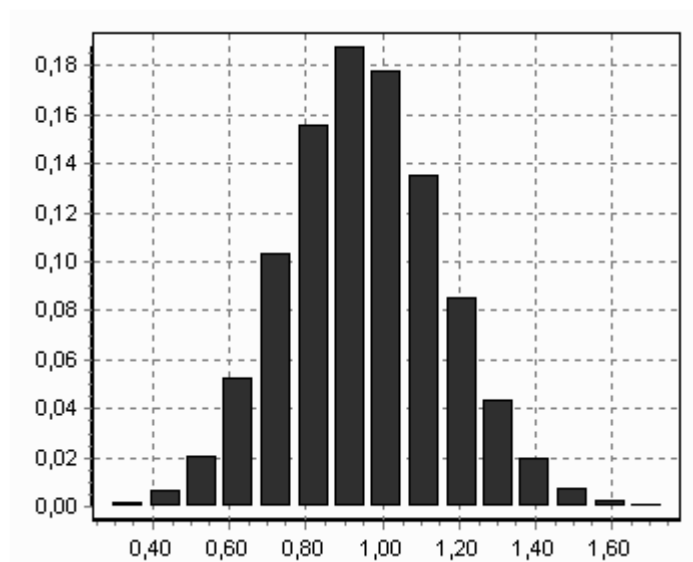


Рисунок 3.4.4 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=22$ дБ, $\sigma=0,2045$, $d(0,66;1,34)$)

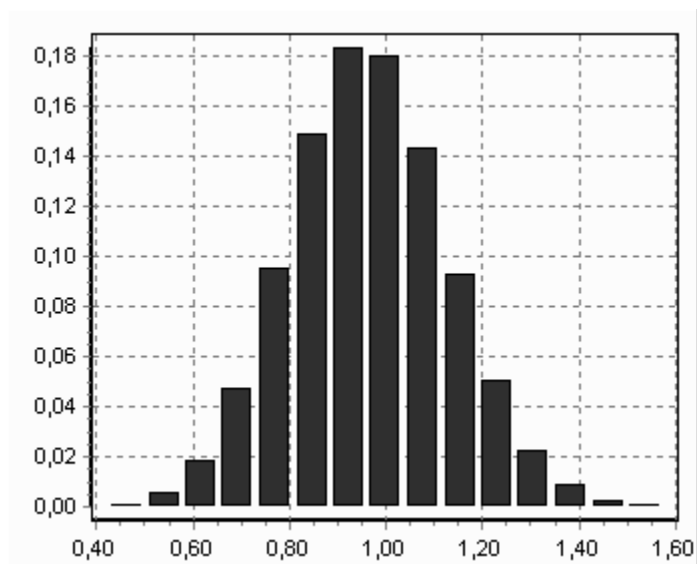


Рисунок 3.4.5 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=24$ дБ, $\sigma=0,1624$, $d(0,73;1,27)$)

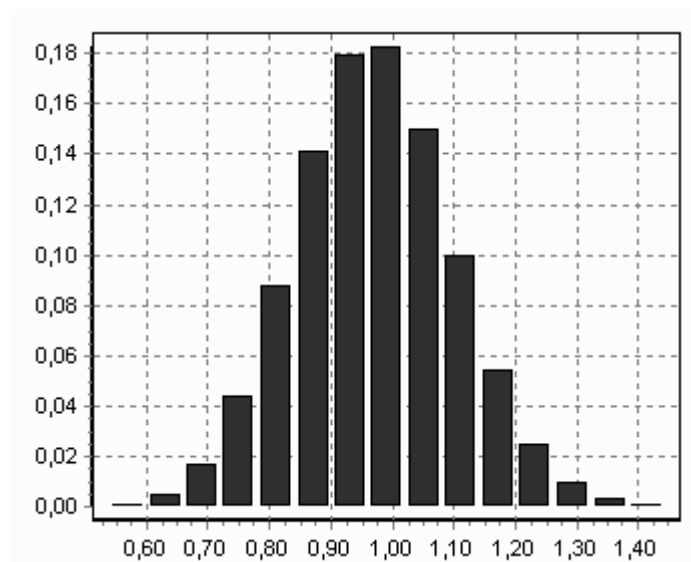


Рисунок 3.4.6 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=26$ дБ, $\sigma=0,1288$, $d(0,79;1,21)$)

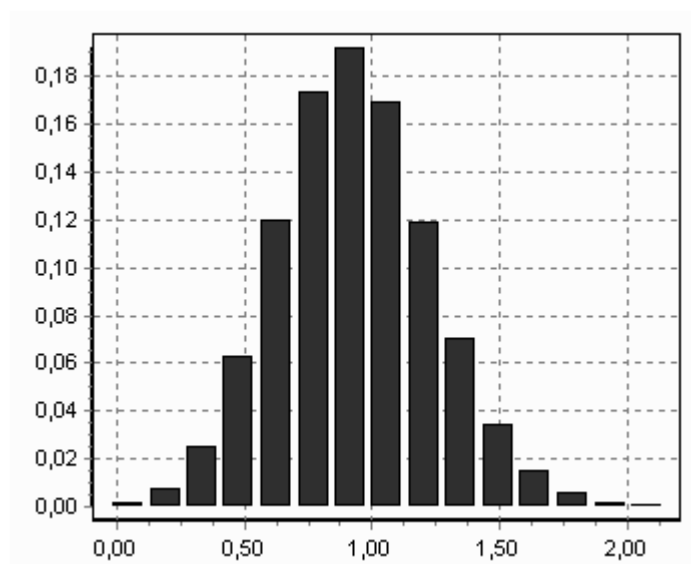


Рисунок 3.4.7 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=18,7$ дБ, $\sigma=0,3008$)

С увеличением отношения сигнал/шум значение σ уменьшается. Найденные значения СКО будут использованы при сравнении алгоритмов.

Как видно из рисунков 3.4.7 и 3.3.2 в стандартном алгоритме тоже значение σ ($\sigma=0,3003$) достигается при $SNR=20$ дБ.

3.5. Сравнение алгоритмов и определение минимального отношения сигнал/шум с учетом помехоустойчивого кодирования для модуляций QAM16, QAM64 и преамбулы состоящей из нескольких символов

Используя результаты моделирования параграфов 3.3 - 3.4 и приложений П2 – ПЗ получим сравнительную таблицу 3.5.1 и рисунок 3.5.1.

Таблица 3.5.1. Зависимость СКО от отношения С/Ш для различных алгоритмов на примере модуляции QPSK.

	С/Ш= 17дБ	С/Ш= 18дБ	С/Ш= 20дБ	С/Ш= 22дБ	С/Ш= 24дБ	С/Ш= 26дБ
Стандартный алгоритм	$\sigma =$ 0,4326	$\sigma =$ 0,3822	$\sigma =$ 0,3003	$\sigma =$ 0,2343	$\sigma =$ 0,1868	$\sigma =$ 0,1468
Модифицированный алгоритм	$\sigma =$ 0,3682	$\sigma =$ 0,3268	$\sigma =$ 0,2591	$\sigma =$ 0,2058	$\sigma =$ 0,1624	$\sigma =$ 0,1288
Стандартный алгоритм увеличенные пилот- поднесущие	$\sigma =$ 0,3774	$\sigma =$ 0,3259	$\sigma =$ 0,2578	$\sigma =$ 0,2038	$\sigma =$ 0,1620	$\sigma =$ 0,1286
Модифицированный алгоритм уменьшенный уровень преамбулы	$\sigma =$ 0,3777	$\sigma =$ 0,3364	$\sigma =$ 0,2666	$\sigma =$ 0,2099	$\sigma =$ 0,1666	$\sigma =$ 0,1323

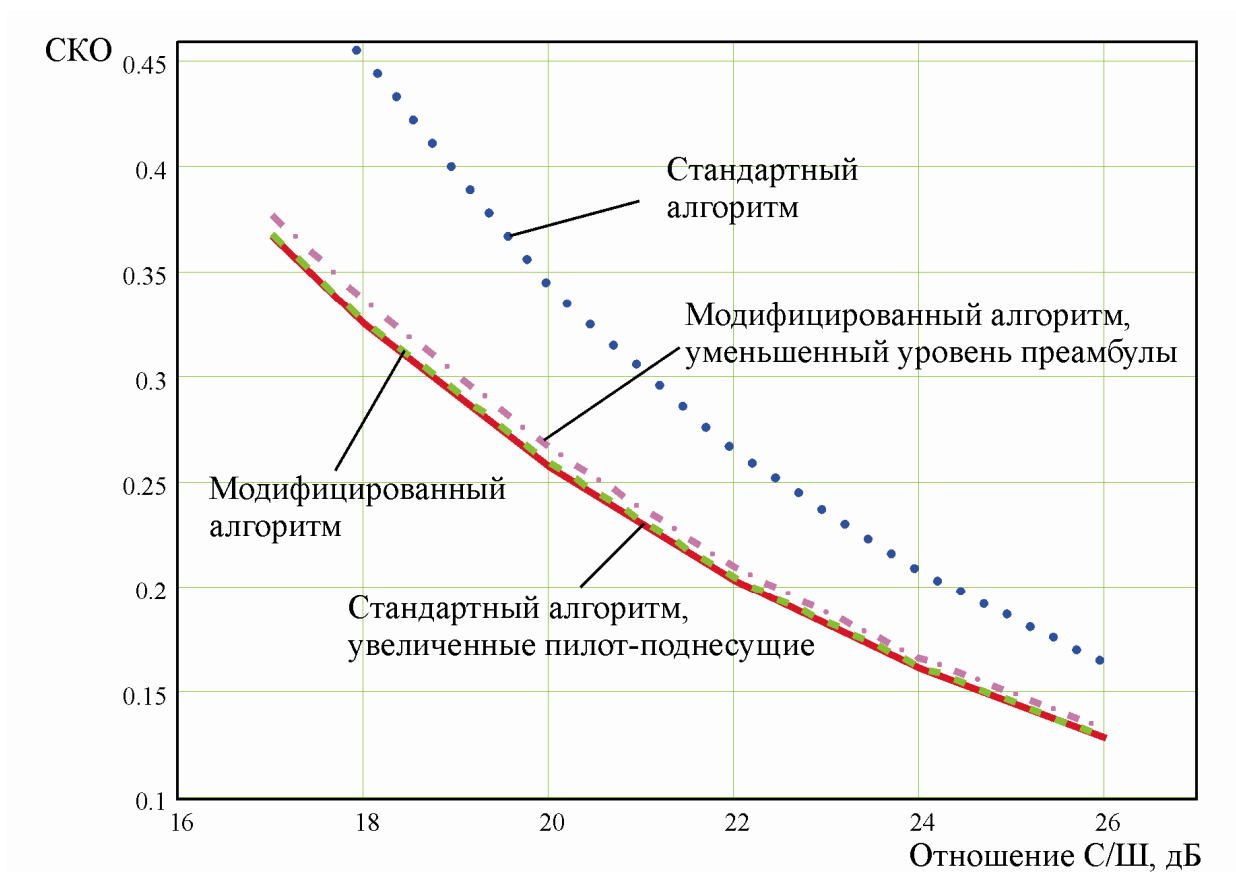


Рисунок 3.5.1 - Зависимость σ от отношения С/Ш для различных алгоритмов

Помехоустойчивое кодирование позволяет передавать сигнал без потерь, в случае если количество ошибок передачи в 2 раза меньше внесенной избыточности. Для примера рассмотрим избыточность 20%, т.е. скорость передачи полезных данных 5/6. В этом случае допустимо 10% ошибок. [76]

Для определения вероятности попадания в заданный интервал в математической статистике используется квантиль — значение, которое заданная случайная величина не превышает с фиксированной вероятностью. [77, 78]

Таблица 3.5.2. Квантили нормального распределения.

Вероятность, %	99,99	99,90	99,00	97,72	97,50	95,00	90,00	84,13
Квантиль	3,715	3,090	2,326	2,000	1,960	1,645	1,282	1,000

Иллюстрация значений квантилей представлена на рисунке 3.5.2. Таким образом, для допустимой вероятности ошибок 10% (вероятность

правильного определения 90%) нужно выбрать интервал от $-1,645\sigma$ до $+1,645\sigma$. Квантиль взята для вероятности 0,95, т.к. итоговая вероятность с учетом одинаковых условий для реальной и мнимой части будет равна $0,95*0,95=0,9025$.

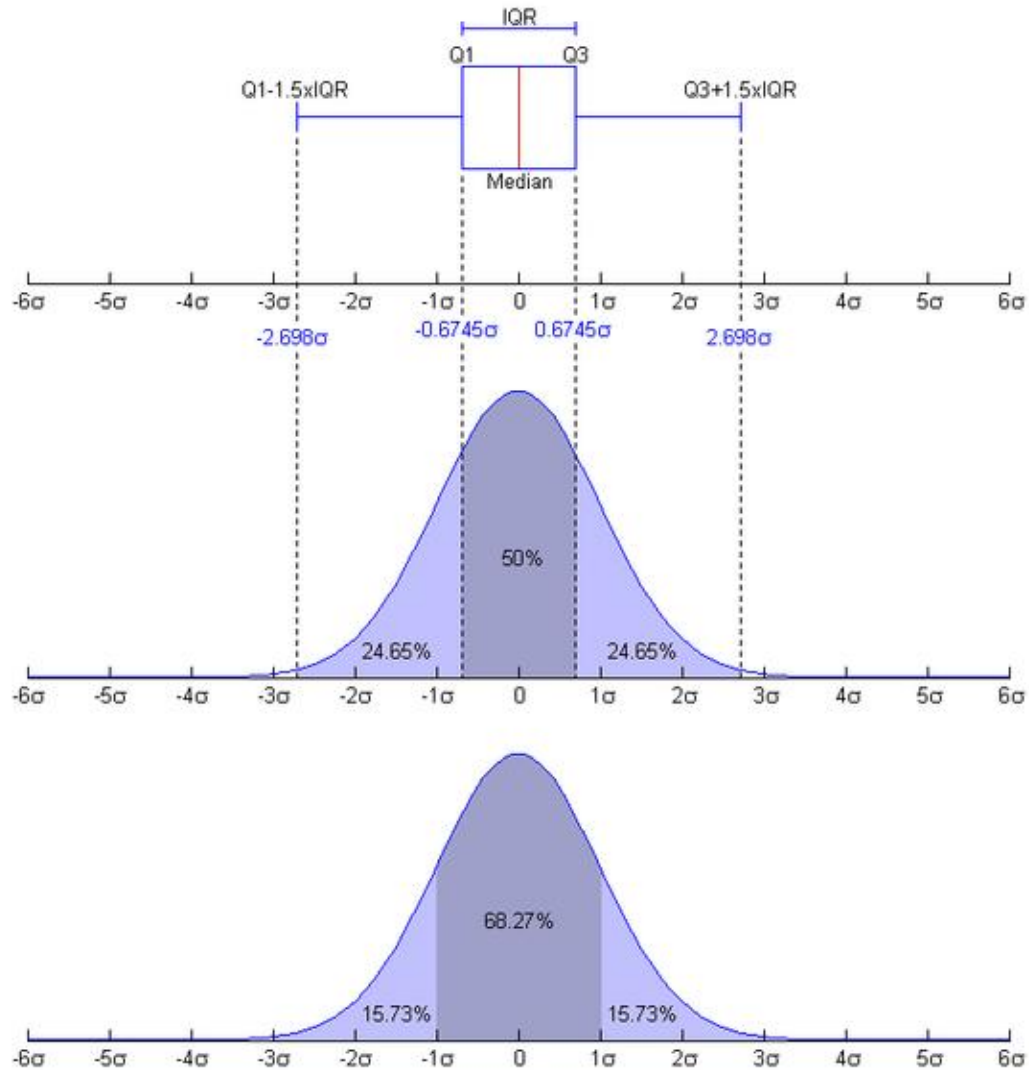


Рисунок 3.5.2 - Квантили нормального распределения

Для модуляции QAM16 и среднего значения максимума равному 1 найдем допустимое максимальное значение СКО так, чтобы 90% определений уровня QAM16 было верным, т.е. от $1-1/3$ до $1+1/3$.

$$\sigma_{16} = (1/3)/1,645 = 0,203$$

Для модуляции QAM64 и среднего значения максимума равному 1 аналогично найдем допустимое максимальное значение СКО.

$$\sigma_{64} = (1/7)/1,645 = 0,087$$

Для повышения точности демодуляции в *OFDM* имеется возможность передавать преамбулу в N символах, уменьшив количество спектральных составляющих в каждом из них. Например, если передавать преамбулу в 2х символах, то количество спектральных компонент в каждом из символов будет уменьшено в 2 раза, соответственно для сохранения мощности нужно поднять уровень этих компонент в $\sqrt{2}$. Уровень шума при таком увеличении сохранится, поэтому погрешность расчета ОККП уменьшится в $\sqrt{2}$. Однако относительный уровень шума в символах данных не изменится, т.к. не изменятся сами символы. Таким образом, можно оценить оптимальное значение разбиения преамбулы на символы. Эта граница обусловлена тем, что минимальная погрешность после применения ОККП будет равна погрешности нахождения спектра символа с данными. СКО в этом случае можно представить как сумму двух независимых среднеквадратических отклонений – обратного комплексного коэффициента передачи и значений спектра символа данных: $s = \sqrt{s_{ОККП}^2 + s_{DATA}^2}$.

Разработана программа моделирования, позволяющая задавать отношение сигнал/шум и задавать различное количество символов в преамбуле. В результате моделирования строится гистограмма распределения уровня принятого *OFDM* символа после применения комплексной передаточной характеристики. Примеры гистограмм для уровня сигнала равного 1 представлены на рисунках 3.5.3 - 3.5.6. (СКО=0,203).

Для найденных значений СКО построим таблицу, показывающую, как разбиение преамбулы на символы позволяет снизить требования к отношению сигнал/шум, при сохранении уровня модуляции и вероятности ошибки. Результаты моделирования представлены в таблицах 3.5.3 и 3.5.4.

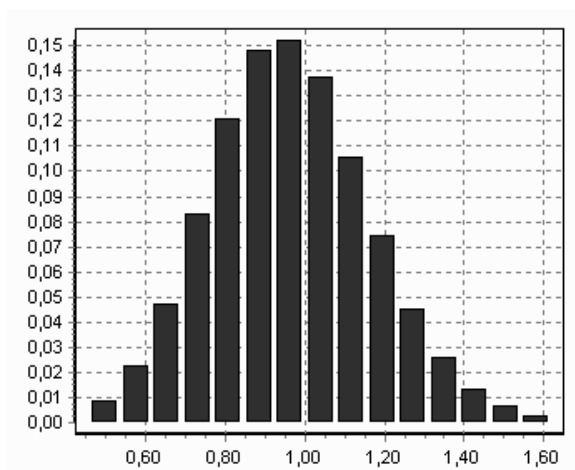


Рисунок 3.5.3 - Распределение уровня
при $C/N=21,3\text{дБ}$, 1 символ в
преамбуле

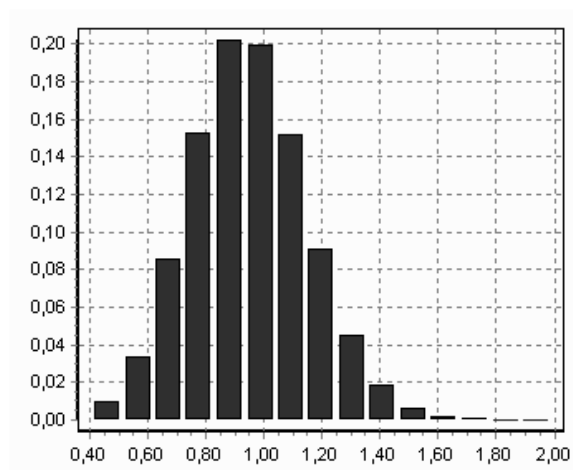


Рисунок 3.5.4 - Распределение уровня
при $C/N=20,2\text{дБ}$, 2 символа в
преамбуле

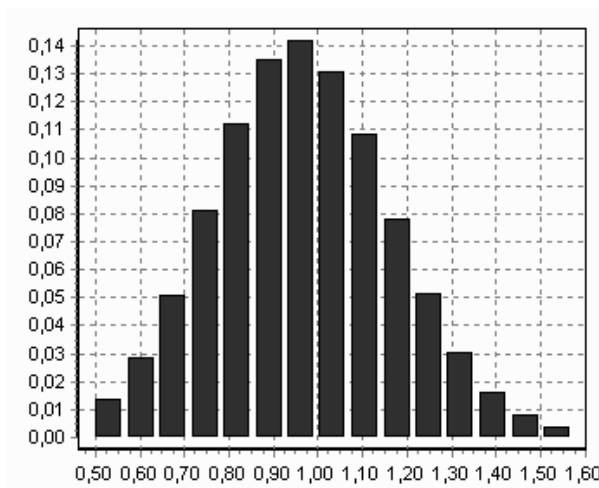


Рисунок 3.5.5 - Распределение уровня
при $C/N=19,2\text{ дБ}$, 8 символов в
преамбуле

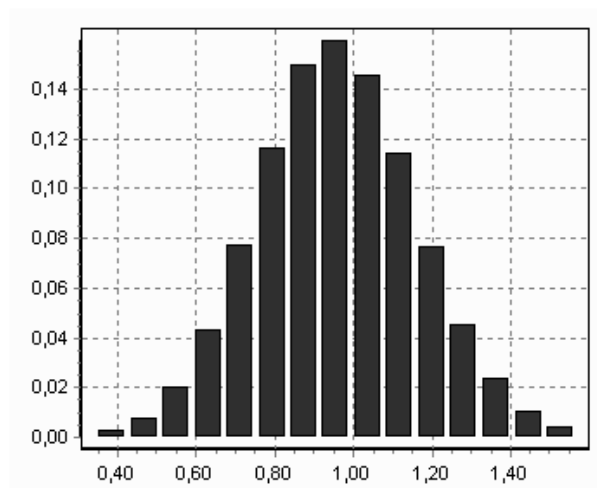


Рисунок 3.5.6 - Распределение уровня
при $C/N=19\text{дБ}$, 16 символов в
преамбуле

Таблица 3.5.3. Минимальное отношение сигнал/шум для QAM16, QAM64.

$\sigma_{16}=0,203$ (QAM16) ~ QAM16 2/3			$\sigma_{64}= 0,087$ (QAM64) ~ QAM64 2/3		
Кол-во символов в преамбуле	Среднее	С/Ш	Кол-во символов в преамбуле	Среднее	С/Ш
1	1	21,3	1	1	28,4
2	1	20,2	2	1	27,4
4	1	19,5	4	1	26,7
8	1	19,2	8	1	26,4
16	1	19	16	1	26,2

Таблица 3.5.4. Зависимость СКО от количества символов в преамбуле на примере модуляции QAM16.

Кол-во символов в преамбуле	С/Ш =16дБ	С/Ш =21 дБ	С/Ш =26 дБ
1	0,38	0,21	0,116
2	0,33	0,183	0,103
4	0,3	0,169	0,095
8	0,29	0,163	0,092
16	0,288	0,161	0,091

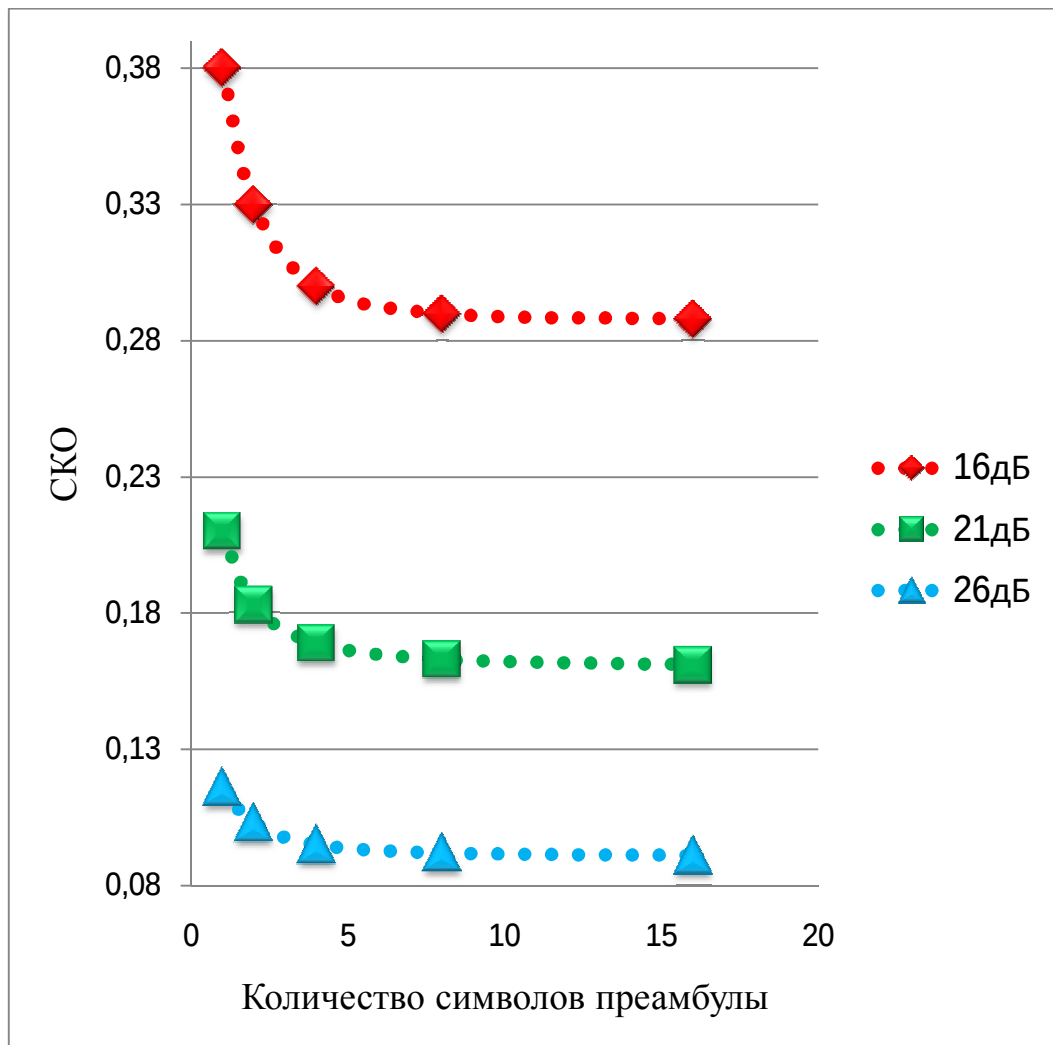


Рисунок 3.5.7 - Зависимость СКО от количества символов преамбулы и уровня шумов

Исходя из результатов таблицы 3.5.3 и рисунка 3.5.7, можно сделать вывод о целесообразности разбиения преамбулы на 2 символа. Такое разбиение по сравнению с 1-м символом позволяет уменьшить отношение С/Ш на 1,1 дБ. Разбиение более, чем на 2 символа приводит к незначительному уменьшению возможного отношения С/Ш. Разбиение на 4 символа по сравнению с 2мя позволяет уменьшить отношение С/Ш на 0,7дБ, на 16 символов – 1,2 дБ. Чем больше символов содержится в преамбуле, тем меньше символов будет отводиться на данные, при той же длительности фрейма.

Экспериментально получена следующая таблица. Обратим внимание, что для символов данных эффективное отношение сигнал/шум зависит от типа модуляции, как показано в параграфе 2.4.

Таблица 3.5.5. Минимальные значения ОСШ в зависимости от применяемой модуляции в *OFDM* символе.

Модуляция	С/Ш для преамбулы	С/Ш для символа данных
<i>QPSK</i> $\frac{1}{2}$	12	4 (-8)
<i>QPSK</i> $\frac{3}{4}$	15,2	7,2 (-8)
<i>QAM16</i> $\frac{1}{2}$	17,3	10,8 (-6.5)
<i>QAM16</i> $\frac{3}{4}$	21,1	14,6 (-6.5)
<i>QAM64</i> $\frac{2}{3}$	27,6	21,3 (-6.3)

3.6. Оценка точности предложенного целочисленного алгоритма коррекции КПХ по сравнению с вещественным алгоритмом

Для моделирования целочисленного алгоритма коррекции КПХ канала связи была разработана программа на *Delphi*, которая позволила произвести отладку целочисленного алгоритма и убедиться в его работоспособности и точности по сравнению с алгоритмом, работающими с вещественными числами.

Данные сравнения СКО уровня поднесущих принятого сигнала от идеального значения после коррекции КПХ сведены в таблицу 3.6.1, а также представлены на рисунке 3.6.1.

На основе представленных данных можно сделать вывод, что переход от вещественных чисел к целым прошел без потери точности коррекции, это означает, что данный алгоритм может быть реализован на языке описания аппаратуры (например, *Verilog*).

Таблица 3.6.1. Зависимость СКО уровня поднесущих от идеального значения для различных соотношений сигнал/шум в АБГШ канале связи

Отношение Сигнал/Шум, дБ	Среднеквадратическое отклонение (СКО)					
	Целочисленный алгоритм			Вещественный алгоритм		
	Битность АЦП			Битность АЦП		
	8 бит	10 бит	12 бит	8 бит	10 бит	12 бит
17	0,3693	0,3688	0,3683	0,3692	0,3678	0,3671
18	0,3295	0,3286	0,3273	0,3291	0,3282	0,3262
20	0,2614	0,2593	0,2591	0,2612	0,2593	0,2585
22	0,2067	0,2056	0,2051	0,2063	0,2056	0,2047
24	0,1641	0,1627	0,1625	0,1637	0,1624	0,1624
26	0,1301	0,1294	0,1289	0,1300	0,1294	0,1287
28	0,1036	0,1027	0,1023	0,1011	0,1026	0,1021
30	0,0827	0,0817	0,0814	0,0820	0,0816	0,0814
32	0,0664	0,0648	0,0644	0,0660	0,0648	0,0644
34	0,0533	0,0515	0,0513	0,0530	0,0514	0,0513
36	0,0432	0,0409	0,0407	0,0429	0,0409	0,0407

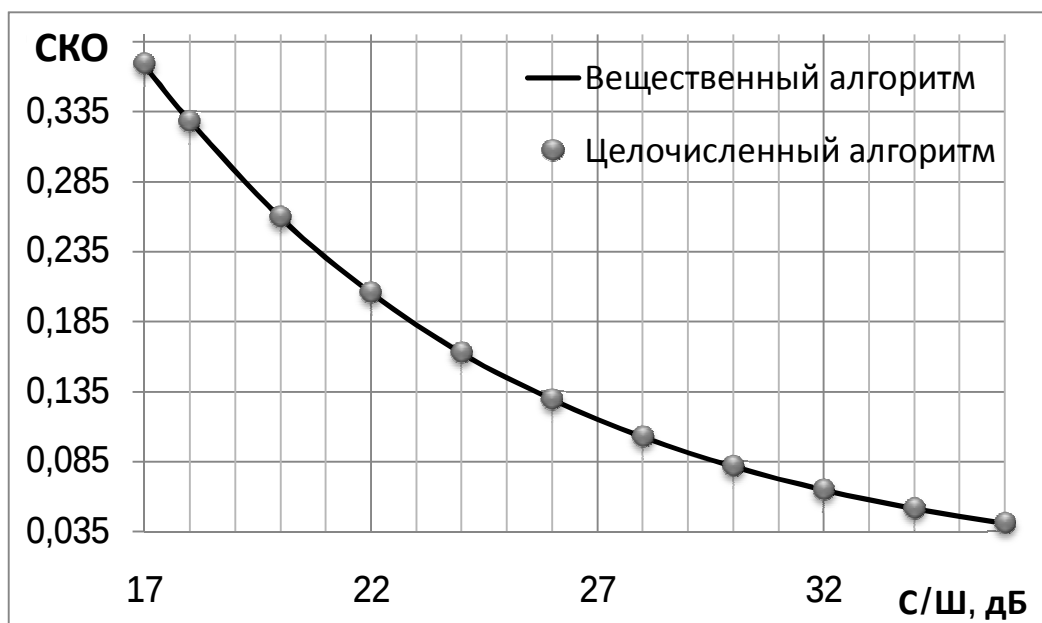


Рисунок 3.6.1- Зависимость СКО уровня поднесущих от идеального значения для различных соотношений сигнал/шум в АБГШ канале связи, АЦП - 10 бит

3.7. Моделирование способа временной синхронизации для преамбулы, имеющей дискретно-непрерывный набор спектральных компонент

Для сравнения способов временной синхронизации, описанных в пункте 1.2. и [40], и предлагаемого способа в одинаковых условиях будем рассматривать преамбулу, содержащую два символа данных, каждый из которых содержит число компонент равное половине используемых спектральных компонент (в описанных в пункте 1.2. способах используется 1000), модулированных по схеме квадратурной фазовой манипуляции (QPSK). Амплитудный спектр преамбулы представлен на рисунке 2.2.1. Количество спектральных компонент в одном символе равно $2z$ и для соответствия способу описанному в [40] $2z=500$. Количество точек в преобразовании Фурье $N=1024$, длительность циклического префикса $N_g=102$. Для моделирования предлагаемого способа использовался первый символ преамбулы. При выбранных параметрах значение порогового коэффициента $k>101$ (получено по результатам моделирования при расчете отношения M_S/A при попадании в массив $r[d]$ N_g дискрет циклического префикса преамбулы без шумов). Для дальнейшего моделирования выберем $k=200$.

Результаты моделирования представленных алгоритмов при параметре $step=1$ и АБГШ канала представлены на рисунке 3.7.1. Каждая точка графика получена по 100 000 реализациям сигнала и шума.

Зависимость дисперсии оценки временного положения сигнала от отношения сигнал/шум для предложенного метода показывает, что помехоустойчивость временной синхронизации значительно увеличилась.

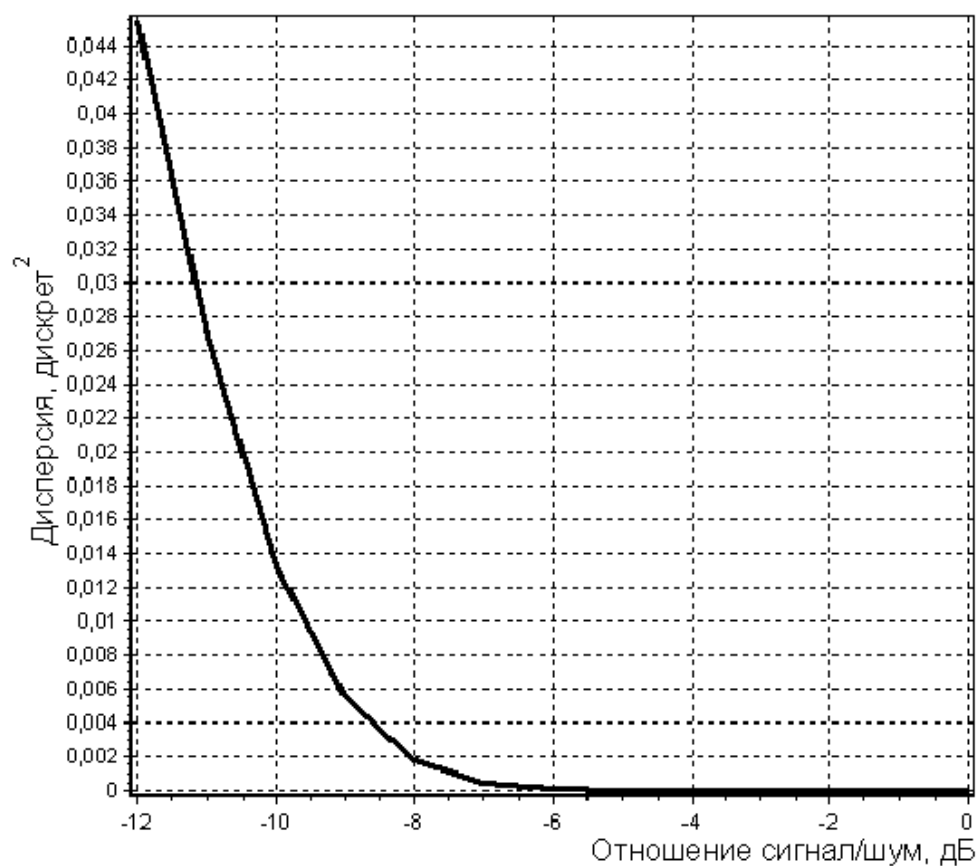


Рисунок 3.7.1 - Зависимость дисперсии оценки временного положения сигнала от отношения сигнал/шум для предложенного способа

Например, при рассмотрении наилучшего метода синхронизации для преамбулы, использующей 1000 спектральных составляющих, имеющей повторяющиеся участки во временной области, представленного в [40] наименьшая дисперсия оценки для АБГШ канала составляет приблизительно $0,8 \text{ дискрет}^2$ при отношении сигнал/шум 0 дБ, тогда как в предложенном способе синхронизации даже при отношении сигнал/шум -12 дБ дисперсия составляет 0,046, а при отношении сигнал/шум -6 дБ – дисперсия равна 0 (см. рисунок 3.7.1), т.е. полностью отсутствуют неправильные определения временного положения преамбулы, таким образом, выигрыш составляет более 12 дБ.

Выводы по главе 3

1. Показано, что гистограммы распределения СКО поднесущих от идеального значения имеют нормальный закон распределения.

2. Показано, что помехоустойчивость модифицированного алгоритма коррекции КПХ на 1,3 дБ выше по сравнению со стандартным. Такой выигрыш обеспечивается повышением уровня пилот-поднесущих в символах данных.

3. Показано, что модифицированный алгоритм использует меньшее количество математических операций. А именно отсутствует перевод в полярную систему координат и обратно в ортогональную перед демодуляцией. Сокращение таких операций упрощает реализацию предложенного алгоритма в аппаратуре, а именно на языках описания аппаратуры *Verilog* или *VHDL*.

4. Показано, что разбиение преамбулы на 2 символа дает хороший выигрыш по помехоустойчивости. Дальнейшее разбиение преамбулы на символы не приводит к столь значительному выигрышу в помехоустойчивости вследствие неизменного уровня спектральных составляющих в символах данных.

5. Показано, что при разбиении преамбулы на 2 символа для работы модуляции QAM16 2/3 требуется отношение С/Ш=20,2 дБ, для QAM64 2/3 - 27,4 дБ. Экспериментально полученная таблица 3.6.1 хорошо совпадает с результатами моделирования, представленными в таблице 3.5.3: Минимальное отношение сигнал шум для модуляции QAM16 1/2 -17,3 дБ; QAM16 3/4 - 21,1 дБ; QAM64 2/3 - 27,6 дБ.

6. Показано, что разработанный целочисленный алгоритм коррекции КПХ системы связи, использующий в расчетах ортогональную систему координат не уступает в точности коррекции алгоритму, работающему с вещественными значениями.

7. Показано, что предложенный способ временной синхронизации для преамбулы имеющей дискретно-непрерывный набор спектральных

компонент, т.е. преамбулы, не имеющей повторяющихся участков во временной области, даже при отношении сигнал/шум -6 дБ позволяет точно определять положение преамбулы без пропусков и ложных обнаружений для выборки из 100 000 реализаций.

8. Показано, что в способах-аналогах наименьшая дисперсия оценки временного положения сигнала для АБГШ канала составляет приблизительно 0,8 дискрет² при отношении сигнал/шум 0 дБ, тогда как в предложенном способе синхронизации даже при отношении сигнал/шум -12 дБ дисперсия составляет 0,046, а при отношении сигнал/шум -6 дБ – дисперсия равна 0, таким образом выигрыш составляет более 12 дБ.

Заключение

1. Показано, что в *OFDM* комплексная передаточная характеристика используется для компенсации искажений. Без такой компенсации система связи становится неработоспособной вследствие невозможности верно демодулировать принятый сигнал. Также показано, что для работы канала связи необходимо произвести временную, тактовую и частотную синхронизацию.

2. Предложен алгоритм коррекции КПХ, позволяющий работать без использования полярной системы координат с использованием преамбулы, имеющей дискретно-непрерывный набор частотных компонент, и который может быть легко реализован в аппаратуре связи за счет использования только целочисленных арифметических операций.

- Показано, что помехоустойчивость алгоритма коррекции КПХ на 1,3 дБ выше по сравнению со стандартным алгоритмом.
- Показано, что алгоритм использует меньшее количество математических операций: отсутствует перевод в полярную систему координат и обратно в ортогональную перед демодуляцией. Это упрощает реализацию предложенного алгоритма в аппаратуре, а именно на языках описания аппаратуры *Verilog* или *VHDL*.
- Показано, что разбиение преамбулы на 2 символа дает значимый выигрыш по помехоустойчивости. Дальнейшее разбиение преамбулы на символы не приводит к столь заметному выигрышу в помехоустойчивости вследствие неизменного уровня спектральных составляющих в символах данных.
- Показано, что разработанный целочисленный алгоритм коррекции КПХ системы связи, использующий в расчетах ортогональную систему координат, не уступает в точности коррекции алгоритму, работающему с вещественными значениями. Моделирование целочисленного алгоритма коррекции КПХ, использующего 32х битные целые числа, при работе в условиях АБГШ канала с отношением сигнал/шум до 36 дБ показало

отсутствие потери точности в сравнении с вещественным алгоритмом коррекции.

- Для разработанного алгоритма коррекции КПХ получены следующие пороговые отношения сигнал/шум, рассчитанные по преамбуле, состоящей из двух символов для модуляций: $QPSK \frac{1}{2}$ - 12 дБ; $QPSK \frac{3}{4}$ - 15,2 дБ; $QAM16 \frac{1}{2}$ - 17,3 дБ; $QAM16 \frac{3}{4}$ - 21,1 дБ; $QAM64 \frac{2}{3}$ - 27,6 дБ.

3. Разработана программа для ЭВМ, позволяющая определить преамбулу, имеющую минимальный пик-фактор, основанная на применении генератора случайных чисел.

4. Предложен способ временной синхронизации, применяющий в качестве решающей функции квадрат модуля комплексной функции взаимной корреляции между принятым сигналом и эталонной преамбулой без циклического префикса. В предложенном способе преамбула, не имеет повторяющихся участков во временной области за счет дискретно-непрерывного набора спектральных компонент. Способ позволил повысить помехоустойчивость за счет резкого сужения области максимума решающей функции.

- Показано, что предложенный способ временной синхронизации даже при отношении сигнал/шум -6 дБ позволяет точно определять положение преамбулы без пропусков и ложных обнаружений для выборки из 100 000 реализаций.
- Показано, что в способах-аналогах наименьшая дисперсия оценки временного положения сигнала для АБГШ канала составляет приблизительно $0,8 \text{ дискрет}^2$ при отношении сигнал/шум 0 дБ, тогда как в предложенном способе синхронизации даже при отношении сигнал/шум -12 дБ дисперсия составляет 0,046, а при отношении сигнал/шум -6 дБ – дисперсия равна 0, таким образом выигрыш по помехоустойчивости составляет более 12 дБ.

5. Рассмотрено применение временной синхронизации для реализации тактовой. Показано, что в случае использования единого опорного

генератора для получения (тактовой) частоты дискретизации ЦАП/АЦП и несущей частоты, выполнение тактовой синхронизации приведет к частотной синхронизации.

6. Созданы на языке высокого уровня программы моделирования для обоснованного выбора рабочих параметров алгоритмов тактовой и временной синхронизации, а также алгоритма коррекции комплексной передаточной характеристики.

7. Результаты работы используются в учебном процессе на кафедре радиотехники и радиосистем ВлГУ. Программы моделирования внедрены в лабораторный практикум по дисциплине «Метрология и радиоизмерения», а также в лабораторные практикумы по дисциплинам «Устройства приема и обработки сигналов», «Дискретная и цифровая обработка информации».

Литература

1. Юрасова, Л.В., Крейнделин, В.Б. Оборудование сетей абонентского радиодоступа семейства стандартов IEEE 802.16: Учебное пособие / МТУСИ. – М., 2007. – 39 с.
2. Рабинер, Л., Гоулд, Б. Теория и применение цифровой обработки сигналов. – М.: Мир, 1978. - 848 с.
3. Скляр, Б. Цифровая связь. Теоретические основы и практическое применение / Б. Скляр. Изд. 2-е, испр.: пер. с англ. – М.: Издательский дом «Вильямс», 2003. – 1104с.
4. Феер, К. Беспроводная цифровая связь. Методы модуляции и расширения спектра: [пер. с англ.] / К. Феер. — М.: Радио и связь, 2000. — 502 с.
5. Прокис, Дж. Цифровая связь. Пер. с англ. / Под ред. Д.Д. Кловского. - М.: Радио и связь, 2000. - 800 с.
6. Бабков, В. Ю. Системы связи с кодовым разделением каналов / В. Ю. Бабков [и др.]. — СПб.: ТРИАДА, 2003. — 293 с.
7. Григорьев, В. К. Системы беспроводного доступа / В.К. Григорьев. — М.: Экотрендз, 2005. — 205 с.
8. Richard van Nee. Basics and History of OFDM – Woodside Networks 2003. Breukelen, The Netherlands.
9. Wang, L. An Overview of peak-to-average power ratio reduction techniques for OFDM systems / L. Wang, C. Tellambura. — Signal Processing and Information Technology, 2006 IEEE International Symposium on, Aug. 2006. - Page(s): 840 - 845.
10. Van Nee, R., Prasad R. OFDM in wireless multimedia communications, L.: Artech House, 2000. - 260 p.
11. Lawrey, E. Multiuser OFDM / Fifth International Symposium on Signal Processing and its Applications, ISSPA `99, Brisbane, Australia, 1999.
12. Hua, Z. Orthogonal Frequency Division Multiplexing for Wireless Communications: Ph.D. thesis. Georgia Institute of Technology, 2004. - 114 p.
13. Бернюков, А.К. Цифровая обработка радиотехнической информации: Практикум / А.К. Бернюков. - Владимир: ВлГТУ, 1994. - 80с.

14. Архипкин, А.В. Стандарт WiMAX: техническое описание, варианты реализации и специфика применения / А.В. Архипкин // Беспроводные технологии. – 2006. - №3. – С. 14-17.
15. Макаров, С. Б. Передача дискретных сообщений по радиоканалам с ограниченной полосой пропускания / С. Б. Макаров, И. А. Цикин. — М.: Радио и связь, 1988. — 304 с.
16. Шахнович, И. Сети городского масштаба: решения рабочей группы IEEE 802.16 - в жизнь! / И.Шахнович //ЭЛЕКТРОНИКА: НТБ, - 2003, - №8, - С. 50.
17. Шахнович, И. Стандарт широкополосного доступа IEEE 802.16 для диапазонов ниже 11 ГГц! / И. Шахнович // ЭЛЕКТРОНИКА: НТБ, - 2005, - №1, - С. 8.
18. Рашич, А.В., Ветров, Ю.В. Помехоустойчивость систем OFDM в условиях действия структурной помехи в части полосы частот сигнала / А. В. Рашич, Ю.В. Ветров // XXXIV Неделя науки СПбГПУ: Материалы Всероссийской межвузовской научно-технической конференции студентов и аспирантов. Ч. VI. — СПб.: Издательство Политехн. университета, - 2006. — С. 25-27.
19. Вишневский, В.М. Широкополосные беспроводные сети передачи информации / В.М. Вишневский, А.И. Ляхов, С.Л. Портной, И.В. Шахнович. – М.: Техносфера, 2005. - 592 с.
20. Вишневский, В.М., Портной, С.Л., Шахнович, И.В. Энциклопедия WiMax. Путь к 4G / В.М. Вишневский, С.Л. Портной, И.В. Шахнович. - М.: Техносфера, 2009. - 472 с.
21. Air Interface for Fixed Broadband Wireless Access System (rollup of 802.16-2001, 802.16a, 802.16c and P802.16d). Institute of Electrical and Electronics Engineers. 2004.
22. IEEE Standard 802.16e-2005, Part 16: Air interface for fixed and mobile broadband wireless access systems, December 2005.
23. Поздняков, В.А., Коробов, Д.С., Поздняков, А.Д., Коротков, С.Ю. Адаптивная алгоритмическая компенсация канальных искажений при демодуляции сигналов OFDM / В.А. Поздняков, Д.С. Коробов, А.Д.

- Поздняков, С.Ю. Коротков // Известия института инженерной физики. - 2012. - №2. - С. 90-95.
24. Вишневский, В. М. Широкополосные беспроводные сети передачи информации / В. М. Вишневский [и др.]. — М.: Техносфера, 2005. — 592 с.
25. Бибило, П.Н. Основы языка VHDL / П.Н. Бибило. Изд. 3-е, доп. — М.: Издательство ЛКИ, 2007. — 328 с.
26. Суворова, Е.А., Шейнин, Ю.Е. Проектирование цифровых систем на VHDL / Е.А. Суворова, Ю.Е. Шейнин. — СПб.: БХВ-Петербург, 2003. — 576 с.
27. Поляков, А.К. Языки VHDL и Verilog в проектировании цифровой аппаратуры / А.К. Поляков. — М.: СОЛОН-Пресс, 2003. — 320 с.
28. Максфилд, К. Проектирование на ПЛИС. Курс молодого бойца / К. Максфилд. — М.: Издательский дом «Додэка-XXI», 2007. — 408 с.
29. Технология DSL [Электронный ресурс] // Режим доступа: <http://www.konturm.ru/tech.php?id=dsl> (дата обращения: 21.04.2014).
30. The Benefits of Discrete Multi-Tone (DMT) Modulation for VDSL Systems [Электронный ресурс] // Режим доступа: www.ikanos.com/wp-content/uploads/2011/12/TWP0001_wo_DN_Benefit-of-DMT1.pdf (дата обращения: 21.04.2014).
31. Специалисты Fujitsu смогли получить скорость передачи данных 100 Гбит/с, используя компоненты, рассчитанные на скорость 10 Гбит/с [Электронный ресурс] // Режим доступа: <http://www.ixbt.com/news/hard/index.shtml?16/62/99> (дата обращения: 22.04.2014).
32. Fujitsu Develops DMT for High-Speed Optical Links [Электронный ресурс] // Режим доступа: <http://www.convergedigest.com/2013/03/fujitsu-develops-dmt-for-high-speed.html> (дата обращения: 23.04.2014).
33. Шахтарин, Б.И. Синхронизация в радиосвязи и радионавигации / Б.И. Шахтарин, И.М. Андрианов, К.С. Калашников, Ю.А. Сидоркина, В.В. Сизых. - М: Горячая линия – Телеком, 2011. - 278 с.

34. Алгоритмы синхронизации в OFDM – системах [Электронный ресурс] // Режим доступа: bmstu-sm5.narod.ru/bumagin/bum_andrew_OFDM.pdf (дата обращения: 20.07.2013).
35. Иванов, А.А. Оптимальная оценка параметров нарушения синхронизации в системе с ортогональным частотным уплотнением / А.А. Иванов // Международная конференция Цифровая обработка сигналов. – 2008. – Выпуск X-1. – С.79-82.
36. Калашников, К.С., Шахтарин, Б.И. Синхронизация OFDM-сигналов во временной и частотной областях / К.С. Калашников, Б.И. Шахтарин // Вестник МГТУ им. Н.Э. Баумана. Сер. «Приборостроение». - 2011. - №1. - С. 18-27.
37. Fusco T. Synchronization Techniques For OFDM Systems: Ph.D. thesis, University of Naples Federico II, 2005. 119 p.
38. Рашич, А.В. Сети беспроводного доступа WiMax / А.В. Рашич. - СПб.: Издательство политехнического университета, 2011 - 180с.
39. Патент США №5,732,113 от 24.03.1998 авторов: Schmidl T.M., Cox D.C., Timing and Frequency Synchronization of OFDM signals.
40. Minn H., Zeng M., Bhargava V. K., On Timing Offset Estimation for OFDM Systems // IEEE COMMUNICATIONS LETTERS, VOL. 4, NO. 7, JULY 2000, pp. 242-244.
41. Диксон, Р.К. Широкополосные системы : [пер. с англ.] / Р.К. Диксон; под общ. ред. В. И. Журавлева. — М.: Связь, 1979. — 304 с.
42. Патент RU 2327281 С2 Устройство для коррекции амплитудно-частотной характеристики трактов связи.
43. Патент РФ № 2433552 Фазовая коррекция для *OFDM* и *MIMO* передачи.
44. Казаков, Л.Н., Кукушкин, Д.С., Исмаилов, А.В. Коррекция фазы *OFDM*-сигналов в условиях полиномиального фазового воздействия и доплеровского рассеяния / Л.Н. Казаков, Д.С. Кукушкин, А.В. Исмаилов // Труды конференции «Цифровая обработка сигналов и ее применение», Москва, - 2008. - С.75-78.

45. Rugini, L., Banelli, P., Leus, G. Simple Equalization of Time-Varying Channels for OFDM // IEEE Communications Letters. 2005. V.9. №7. P.619-621.
46. Bahai, A.R.S., Saltzberg, B.R. Multi-carrier Digital Communications: Theory and Applications of OFDM. N.Y.: Kluwer Academic Publishers, 2002. - 216 p.
47. Безруков, В.Н., Комаров, П.Ю., Коржихин, Е.О. Специфика коррекции характеристик радиоканала в системе цифрового телевидения по стандарту 24 DVB-T / В.Н. Безруков, П.Ю. Комаров, Е.О. Коржихин // Труды Московского технического университета связи и информатики – М.: «ИД Медиа Паблишер», – 2008. – Т1. – С. 437-440.
48. Ермолаев, В.Т. Флакман, А.Г. Теоретические основы обработки сигналов в системах мобильной радиосвязи. Электронное методическое пособие. – Нижний Новгород, 2010. – 107 с.
49. Позднякова, Л.В. Свидетельство о государственной регистрации программы для ЭВМ №2015618433: «Программа моделирования коррекции комплексной передаточной характеристики канала связи по преамбуле и пилот-поднесущим в соответствии со стандартом WiMAX 802.16.2004».
50. Корн, Г., Корн, Т. Справочник по математике для научных работников и инженеров. / Г. Корн, Т. Корн. - М.: Изд. Наука, 1973. - 832 с.
51. Письменный, Д.Т. Конспект лекций по высшей математике: полный курс / Д.Т. Письменный – 5-е изд. – М.: Айрис-пресс, 2007. – 608с.
52. Поздняков, В.А., Позднякова, Л.В., Карпов, И.В., Коробов, Д.С. Особенности коррекции комплексной передаточной характеристики по преамбуле OFDM сигнала / В.А. Поздняков, Л.В. Позднякова, И.В. Карпов, Д.С. Коробов // Проектирование и технология электронных средств. – 2013. - №2. – С. 5-8.
53. Позднякова, Л.В., Поздняков, В.А. Технология OFDM: методы коррекции комплексной передаточной характеристики канала связи / Л.В. Позднякова, В.А. Поздняков // Сборник трудов XI международной научно-технической конференции «Перспективные технологии в средствах передачи информации», Владимир - Суздаль, - 2015. – С. 92-95.

54. Позднякова, Л.В., Поздняков, В.А. Повышение точности коррекции комплексной передаточной характеристики по преамбуле OFDM сигнала / Л.В. Позднякова, В.А. Поздняков // Проектирование и технология электронных средств. – 2014 . - №4. – С.14-18.
55. Поздняков, В.А., Позднякова, Л.В., Карпов, И.В. Определение комплексной передаточной характеристики канала связи *OFDM* при разделении испытательного линейчатого спектра преамбулы на несколько символов / В.А. Поздняков, Л.В. Позднякова, И.В. Карпов // Радиотехнические и телекоммуникационные системы. – 2012. - №3 – С. 49-53.
56. Позднякова, Л.В. Целочисленный алгоритм коррекции комплексной передаточной характеристики канала связи OFDM / Л.В. Позднякова // Радиотехнические и телекоммуникационные системы. – 2015. - №1. – С.70-77.
57. Позднякова, Л.В., Поздняков, В.А. Технология OFDM: целочисленный алгоритм коррекции комплексной передаточной характеристики / Л.В. Позднякова, В.А. Поздняков // Сборник трудов XI международной научно-технической конференции «Перспективные технологии в средствах передачи информации», Владимир - Суздаль, - 2015. – С. 95-97.
58. Позднякова, Л.В. Свидетельство о государственной регистрации программы для ЭВМ №2015618752: «Программа моделирования целочисленного алгоритма коррекции комплексной передаточной характеристики канала связи на основе ортогонального частотного разделения каналов с мультиплексированием (OFDM)».
59. Кантор, Л.Я. и др. Спутниковая связь и вещание. Справочник. 2-е изд., перераб. и доп. - М.: Радио и связь, 1988. - 344 с.
60. Рашич, А.В. Применение блочного кодирования для снижения пик-фактора сигналов с OFDM / А. В. Рашич, С. Б. Макаров // Труды СПбГТУ № 507. - СПб. : Изд-во Политехнического университета, - 2008. — С. 170-178.
61. Поздняков, В.А., Позднякова, Л.В., Никитин, О.Р. Алгоритм снижения пик-фактора преамбулы для OFDM и DMT сигналов / В.А. Поздняков, Л.В.

Позднякова, О.Р. Никитин // Проектирование и технология электронных средств. – 2013. - №4. – С. 23-26.

62. Рашич, А.В. Снижение пик-фактора сигналов с ортогональным частотным уплотнением / А. В. Рашич, С. Б. Макаров // Научно-технические ведомости СПбГПУ № 2(55)/2008. — СПб.: Изд-во Политехнического университета, - 2008. — С. 79-84.

63. Рашич, А.В. Снижение пик-фактора OFDM-сигналов с помощью блочного кодирования / А. В. Рашич, С. Б. Макаров, Д. В. Салюк // 12-я Санкт-Петербургская международная конференция Межрегиональная информатика-2008. Труды конференции. - СПб.: СПОИСУ, - 2008. - С. 141-146.

64. Копысов, А.Н., Мошонкин, В.А., Загидуллин, Ю.Т. Исследование алгоритмов снижения пик-фактора сигнальных конструкций на базе дискретно-частотных сигналов / А.Н. Копысов, В.А. Мошонкин, Ю.Т. Загидуллин // Материалы VIII Международной научно-практической конференции электронные средства и системы управления. - 2012. - С. 18-22.

65. Сюваткин, В.С. WiMAX — технология беспроводной связи: основы теории, стандарты, применение / В.С. Сюваткин. — СПб.: БХВ-Петербург, 2005. — 368 с.

66. Позднякова, Л.В. Свидетельство о государственной регистрации программы для ЭВМ №2015618979: «Программа расчета коэффициентов преамбулы для системы связи на основе ортогонального частотного разделения каналов с мультиплексированием (OFDM) для поиска ее оптимального по пик-фактору варианта».

67. Поздняков, В.А., Поздняков, А.Д., Коробов, Д.С. Проектирование и моделирование алгоритма расчета комплексного коэффициента передачи канала связи при демодуляции сигналов OFDM / В.А. Поздняков, А.Д. Поздняков, Д.С. Коробов // Проектирование и технология электронных средств. – 2012. - №4 – С. 43-49.

68. Позднякова, Л.В., Поздняков, В.А. Заявка на изобретение 2015109441: «Способ временной синхронизации системы связи на основе ортогонального

частотного разделения каналов с мультиплексированием по преамбуле». Опубликовано 10.10.2016, Бюл. 28.

69. Поздняков, В.А., Карпов, И.В. Временная синхронизация и адаптивное определение длины циклического префикса в канале связи OFDM / В.А. Поздняков, И.В. Карпов // Проектирование и технология электронных средств. – 2013. - №1 – С. 12-15.

70. Поздняков, В.А., Позднякова, Л.В. Никитин, О.Р. Генератор испытательных радиосигналов на основе микроконтроллера / В.А. Поздняков, Л.В. Позднякова, О.Р. Никитин // Радиотехнические и телекоммуникационные системы. – 2011. - №1. – С. 21-24.

71. Тихонов, В.И. Статистическая радиотехника / В. И. Тихонов. — 2-е изд., перераб. и доп. — М.: Радио и связь, 1982. — 624 с.

72. Позднякова, Л.В., Поздняков, В.А., Никитин, О.Р. Генератор испытательных аналоговых радиосигналов с возможностью генерации нормального шума в заданной полосе частот / Л.В. Позднякова, В.А. Поздняков, О.Р. Никитин // Методы и устройства передачи и обработки информации. - Муром. — 2012. - №1. – С. 19–22.

73. Фленов, М.Е. Библия Delphi / М.Е. Фленов. – СПб.: БХВ - Петербург, 2004. – 880с.

74. Возневич, Э. Delphi. Освой самостоятельно: Пер. с англ. / Э. Возневич. – М.: Восточная Книжная Компания, 1996. – 736 с.

75. Позднякова, Л.В., Поздняков, А.Д. Алгоритмы формирования OFDM испытательного сигнала с заданными параметрами модуляции и шума / Л.В. Позднякова, А.Д. Поздняков // Проектирование и технологии электронных средств. - Владимир. – 2012. – №4. – С. 31–34.

76. Коржик, В.И., Финк, Л.М. Помехоустойчивое кодирование дискретных сообщений в каналах со случайной структурой / В.И. Коржик, Л.М. Финк. – М.: Связь, 1975. – 272 с.

77. Орлов, А.И. Математика случая: Вероятность и статистика - основные факты / А.И. Орлов. Учебное пособие. – М.: МЗ-Пресс, 2004. – 110 с.

78. Айвазян, С. А. и др. Прикладная статистика: Исследование зависимостей: Справ. изд. / С. А. Айвазян, И. С. Енюков, Л. Д. Мешалкин; Под ред. С. А. Айвазяна. — М.: Финансы и статистика, 1985. — 487 с.

П.1. Алгоритмы работы программы моделирования

Краткое описание переменных используемых в алгоритме.

mod_type (определяет вид модуляции),

n_komponent (кол-во активных компонент в *OFDM* символе),

n_byte_ofdm (кол-во байт в *OFDM* символе), *ind*, *i*, *j*, *k* (индексы),

ofdm_symbols_count (кол-во *OFDM* символов для генерации),

n_DAC_Bits (битность ЦАП),

m (кол-во столбцов гистограммы, вычисляется по формуле $3.3\log N$ [7]),

f1 (нижняя граница полосы пропускания),

f2 (верхняя граница полосы пропускания),

kol_tochek (кол-во точек для построения шума),

granica1 (индекс), *granica2* (индекс),

kol_tochek_signal (кол-во точек для построения сигнала),

f_disk (частота дискретизации),

window_type (переменная для определения используется окно для АЧХ или нет),

window_type_shum (переменная для определения используется окно для построения шума или нет),

x1 (минимум шума), *x2* (максимум шума),

skz (скз для создания определенного уровня шума)

n (размерность *FFT*),

L (сдвиг от начала символа при считывании данных)

n_cp (длина циклического префикса *OFDM* символа),

pr (количество символов преамбулы),

n_steps_mod (кол-во шагов за которое будет записан 1 байт данных),

noiseIndexMax (верхняя граница шума),

noiseIndexMin (нижняя граница шума): **целые числа;**

sigma (сигма),

step (шаг для построения закона распределения шума),

z (переменная для построения закона распределения шума),

sr_z (среднее значение шума),

shumReOne (одна точка шума для реальной части сигнала),

shumImOne (одна точка шума для мнимой части сигнала),

b_bit (коэффициент для задания уровня сигнала в соответствии с битностью ЦАП): **вещественные числа;**

s_BPSK - сигнальные созвездия из 8ми комплексных составляющих для кодирования одного байта.

s_QPSK - сигнальные созвездия из 4х комплексных составляющих для кодирования одного байта.

s_QAM16 - сигнальные созвездия из 2х комплексных составляющих для кодирования одного байта.

s_QAM64 - сигнальные созвездия из 4х комплексных составляющих для кодирования 3х байт.

shum (шум),

shumRe (шум для реальной части сигнала),

shumIm (шум для мнимой части сигнала),

x_reACHH (АЧХ реальная часть), *x_imACHH* (АЧХ мнимая часть),

filterRe (массив для фильтрации реальной части шума),

filterIm (массив для фильтрации мнимой части шума),

windowACHH (окно для АЧХ),

window (окно для шума),

shum_window (шум с окном),

sRe_ofdm (OFDM символ без циклического префикса, реальная часть),

sIm_ofdm (OFDM символ без циклического префикса, мнимая часть),

cpRe (циклический префикс, реальная часть),

cpIm (циклический префикс, мнимая часть),

sRe_ofdm_itog (OFDM символ с циклическим префиксом, реальная часть),

sIm_ofdm_itog (OFDM символ с циклическим префиксом, мнимая часть): **массивы вещественных чисел;**

specRe (реальная часть спектра), *specIm* (мнимая часть спектра),

in_data (исходные данные): **массив целых чисел;**

noise (переменная для возвращения значения шума в определенной точке): **массив типа *noise_t*** (тип *noise_t*=запись: *noise*: вещественное число, *isUsed*: булевый тип);

Начало

1. *noiseIndex*=0;

noiseIndexMin=0;

kol_tochek=4000;

kol_tochek_fft=2048;

noiseIndexMax= *kol_tochek_fft*-1;

ofdm_symbols_count=1;

n=256;

n_komponent=192;

$m=11$;

L , f_{disk} , pr - данные берутся из полей программы;

2. если задана полоса пропускания, то переход к п. 3, иначе к п. 4;

3. нижняя ($f1$) и верхняя ($f2$) границы полосы пропускания задаются из соответствующих полей программы, а также задаются

$noiseIndex=512$;

$noiseIndexMin=512$;

$noiseIndexMax=2047-512$;

4. $f1=0$;

$f2=f_{disk}/2$;

5. в зависимости от того, что введено в поле «Циклический префикс», задается параметр n_{cp} ($n/4$, $n/8$, $n/16$, $n/32$ или $n/64$);

6. если выбрана модуляция $BPSK$, тогда переход к п. 7, иначе переход к п. 8;

7. $n_{byte_ofdm}=n_{komponent}*1/8$;

$n_{steps_mod}=8$, за 8 шагов будет записан 1 байт, 1 созвездие переносит 1 бит данных;

8. если выбрана модуляция $QPSK$, тогда переход к п. 9, иначе переход к п. 10;

9. $n_{byte_ofdm}=n_{komponent}*2/8$, 2 - количество бит на 1 спектральную составляющую, делим на 8 чтобы перейти от бит к байтам;

$n_{steps_mod}=4$, за 4 шага будет записан 1 байт, 1 созвездие переносит 2 бита данных;

10. если выбрана модуляция $QAM16$, тогда переход к п. 11, иначе переход к п. 12;

11. $n_{byte_ofdm}=n_{komponent}*4/8$, 4-количество бит на 1 спектральную составляющую, делим на 8 чтобы перейти от бит к байтам;

$n_{steps_mod}=2$, за 2 шага будет записан 1 байт, 1 созвездие переносит 4 бита данных;

12. если выбрана модуляция $QAM64$, тогда переход к п. 13, иначе переход к п. 14;

13. $n_{byte_ofdm}=n_{komponent}*6/8$, 6-количество бит на 1 спектральную составляющую, делим на 8 чтобы перейти от бит к байтам;

$n_{steps_mod}=4$, за 4 шага будет записано 3 байта, 1 созвездие переносит 6 бит данных;

14. в зависимости от того, что введено в поле «Битность ЦАП», задается параметр n_DAC_Bits (8, 10, 12, 14 или 16);
15. если $n_DAC_Bits=8$, переход к п. 16, иначе к п. 17;
16. $skz = 64; b_bit=9/1,1007$;
17. если $n_DAC_Bits=10$, переход к п. 18, иначе к п. 19;
18. $skz = 255; b_bit=36/1,1007$;
19. если $n_DAC_Bits=12$, переход к п. 20, иначе к п. 21;
20. $skz = 1018; b_bit=144/1,1007$;
21. если $n_DAC_Bits=14$, переход к п. 22, иначе к п. 23;
22. $skz = 4087; b_bit=578/1,1007$;
23. если $n_DAC_Bits=16$, переход к п. 24, иначе к п. 25;
24. $skz = 16362; b_bit=2314/1,1007$;
25. $sigma=skz/10^{(C/Ш)/20}$;
26. $sigma = sigma \sqrt{\frac{f_disk}{2(f_2 - f_1)}}$;
27. запуск процедуры *getShum* для формирования массива шума (*shum*);
28. если используется окно при построении шума, тогда переход к п. 29, иначе к п. 30;
29. $window_type_shum=2$;
30. $window_type_shum=1$;
31. запуск процедуры *CreateWindow* для создания массива окна – *window*;
32. $sr_z=0$;
33. $i=0$;
34. если $i < kol_tochek-1$, тогда переход к п. 35, иначе переход к п. 37;
35. $sr_z=sr_z+shum[i]$;
36. $i=i+1$, переход к п. 34;
37. $sr_z=sr_z/kol_tochek$;
38. $i=0$;
39. если $i < kol_tochek-1$, тогда переход к п. 40, иначе переход к п. 42;
40. $shum[i] = shum[i]-sr_z$, вычитаем среднее значение из шума;
41. $i=i+1$, переход к п. 39;
42. $i=0$;
43. если $i < kol_tochek_fft-1$, тогда переход к п. 44, иначе переход к п. 46;
44. $shum_window[i] = shum[i] * window[i]$;

45. $i=i+1$, переход к п. 73;
46. $i=0$;
47. если $i < kol_tochek_fft-1$, тогда переход к п. 48, иначе переход к п. 50;
48. $shumRe[i]=shum_window[i]$, получение шума для реальной части сигнала;
49. $i=i+1$, переход к п. 47;
50. повторение п. 27, 32-49 для получения массива шума ($shumIm$) для мнимой части сигнала, переход к п. 51;
51. если используется окно для импульсной характеристики, тогда переход к п. 52, иначе к п. 53;
52. $window_type=2$;
53. $window_type=1$;
54. если используется фильтрация нормально распределенного шума, тогда переход к п. 55, иначе к п. 56;
55. запуск процедуры *FiltrNormalNoise* для получения фильтрованных массивов шума $shumRe$, $shumIm$;
56. если $window_type_shum=2$ (если использовалось окно при построении шума), тогда переход к п. 57, иначе к п. 62;
57. $i = noiseIndexMin$;
58. если $i < noiseIndexMax$, тогда переход к п. 59, иначе переход к п. 62;
59. $shumRe[i] = shumRe[i] / window[i]$;
60. $shumIm[i] = shumIm[i] / window[i]$;
61. $i=i+1$, переход к п. 58;
62. запуск процедуры *GetPreamble* для получения реальной и мнимой частей спектра преамбулы $preambRe$, $preambIm$;
63. $j=0$;
64. $k=0$;
65. если $k < pr-1$, тогда переход к п. 66, иначе к п. 75;
66. $start = 1+round(100/pr)*k$, $stop = start + round(100/pr)-1$;
67. запуск процедуры *makePreamble* для получения реальной и мнимой частей преамбулы во временной области $preambReItog$, $preambImItog$, а также для получения эталонных спектров преамбулы $preambReStand$, $preambImStand$;
68. $i=0$;
69. если $i < n+n_cp-1$, тогда переход к п. 70, иначе к п. 73

70. запуск процедуры *GetNoiseVal* для получения одной точки шума для реальной и для мнимой частей преамбулы *shumReOne*, *shumImOne*;
71. $preambReItog[i] = preambReItog[i] + shumReOne$,
 $preambImItog[i] = preambImItog[i] + shumImOne$;
72. $i = i + 1$, переход к п. 69;
73. запуск процедуры *getOKKP* для определения реальной и мнимой частей обратного комплексного коэффициента передачи *OKKP_Re*, *OKKP_Im*.
74. $k = k + 1$, переход к п. 65;
75. если в строке «Вид модуляции» выбрана модуляция *BPSK*, тогда переход к п. 76, иначе переход к п. 77;
76. $mod_type = cBPSK$;
77. если в строке «Вид модуляции» выбрана модуляция *QPSK*, тогда переход к п. 78, иначе переход к п. 79;
78. $mod_type = cQPSK$;
79. если в строке «Вид модуляции» выбрана модуляция *QAM16*, тогда переход к п. 80, иначе переход к п. 81;
80. $mod_type = cQAM16$;
81. если в строке «Вид модуляции» выбрана модуляция *QAM64*, тогда переход к п. 82, иначе переход к п. 83;
82. $mod_type = cQAM64$;
83. $ind = 0$;
84. если $ind < ofdm_symbols_count - 1$, тогда переход к п. 85, иначе переход к п. 91;
85. $i = 0$;
86. если $i < n_byte_ofdm - 1$, тогда переход к п. 87, иначе переход к п. 89;
87. массив *in_data[i]* заполняется входными данными;
88. $i = i + 1$, переход к п. 86;
89. запуск процедуры *CreateOneSymbolOFDM* для формирования 1 *OFDM* символа, получение реальной (*signalReItog*) и мнимой (*signalImItog*) частей сигнала с наложенным шумом, а также спектров (*specRe*, *specIm*) и эталонных значений пилот-поднесущих (*PilotReIdeal*, *PilotImIdeal*);
90. $ind = ind + 1$, переход к п. 84;
91. $j = 0$;
92. $i = L$;
93. если $i < L + n - 1$, то переход к п. 94, иначе к п. 96;

94. получение данных с момента L
 $S_RePrin[j]=signalReItog[i]$, $S_ImPrin[j]=signalImItog[i]$;

95. $j = j+1$, $i=i+1$, переход к п. 93;

96. запуск процедуры *FFT* для получения спектров принятого символа S_RePrin , S_ImPrin ;

97. $i=0$;

98. если $i<100$, тогда переход к п. 99, иначе к п. 101;

99. Комплексное умножение ОККП на символ с данными для получения спектров скорректированных данных
 $S_RePrin_okkp[i]=S_RePrin[i]*OKKP_Re[i]-S_ImPrin[i]*OKKP_Im[i]$,
 $S_ImPrin_okkp[i]=S_ImPrin[i]*OKKP_Re[i]+S_RePrin[i]*OKKP_Im[i]$;

100. $i=i+1$, переход к п. 98;

101. $i=156$;

102. если $i<n$, тогда переход к п. 103, иначе к п. 105;

103. Комплексное умножение ОККП на символ с данными для получения спектров скорректированных данных
 $S_RePrin_okkp[i]=S_RePrin[i]*OKKP_Re[i]-S_ImPrin[i]*OKKP_Im[i]$,
 $S_ImPrin_okkp[i]=S_ImPrin[i]*OKKP_Re[i]+S_RePrin[i]*OKKP_Im[i]$;

104. $i=i+1$, переход к п. 102;

105. $j=0$, $i=2$;

106. если $i \leq 98$, то переход к п. 107, иначе к п. 109;

107. запись в массивы *PilotRePrin* и *PilotImPrin* реальной и мнимой частей пилот-поднесущих:
 $PilotRePrin[j]= S_RePrin_okkp[i]$;
 $PilotImPrin[j]= S_ImPrin_okkp [i]$;

108. $i=i+32$, $j=j+1$, переход к п. 106;

109. $j=4$, $i=158$;

110. если $i \leq 254$, то переход к п. 111, иначе к п. 113;

111. запись в массивы *PilotRePrin* и *PilotImPrin* реальной и мнимой частей пилот-поднесущих:
 $PilotRePrin[j]= S_RePrin_okkp[i]$;
 $PilotImPrin[j]= S_ImPrin_okkp [i]$;

112. $i=i+32$, $j=j+1$, переход к п. 110;

113. $j=2$, $i=0$;

114. если $i \leq 3$, то переход к п. 115, иначе к п. 117;

115. расчет реальной и мнимой частей ОКПХ по пилот-поднесущим:

$znam = PilotRePrin[i] * PilotRePrin[i] + PilotImPrin[i] * PilotImPrin[i];$
 $PilotReOKKP[j] = (PilotReIdeal[j] * PilotRePrin[i] + PilotImPrin[i] * PilotImIdeal[j]) / znam;$
 $PilotImOKKP[j] = (PilotImIdeal[j] * PilotRePrin[i] - PilotImPrin[i] * PilotReIdeal[j]) / znaml;$

116. $i = i + 1, j = j + 32$, переход к п. 114;

117. $j = 158, i = 4$;

118. если $i \leq 7$, то переход к п. 119, иначе к п. 121;

119. расчет реальной и мнимой частей ОКПХ по пилот-поднесущим:

$znam = PilotRePrin[i] * PilotRePrin[i] + PilotImPrin[i] * PilotImPrin[i];$
 $PilotReOKKP[j] = (PilotReIdeal[j] * PilotRePrin[i] + PilotImPrin[i] * PilotImIdeal[j]) / znam;$
 $PilotImOKKP[j] = (PilotImIdeal[j] * PilotRePrin[i] - PilotImPrin[i] * PilotReIdeal[j]) / znaml;$

120. $i = i + 1, j = j + 32$, переход к п. 118;

121. $i = 2$;

122. если $i \leq 66$, то переход к п. 123, иначе к п. 139;

123. нахождение коэффициентов для аппроксимации:

$factor_k = (PilotReOKKP[i + 32] - PilotReOKKP[i]) / 32;$
 $factor_b = PilotReOKKP[i] - factor_k * i;$

124. если $i = 2$, то переход к п. 125, иначе к п. 129;

125. $j = 0$;

126. если $j < 2$, то переход к п. 127, иначе к п. 129;

127. аппроксимация для индексов меньше 2

$PilotReOKKP[j] = factor_k * j + factor_b;$

128. $j = j + 1$, переход к п. 126;

129. $j = i + 1$;

130. если $j \leq i + 31$, то переход к п. 131, иначе к п. 133;

131. аппроксимация $PilotReOKKP[j] = factor_k * j + factor_b;$

132. $j = j + 1$, переход к п. 130;

133. если $i = 66$, то переход к п. 134, иначе к п. 138;

134. $j = i + 33$;

135. если $j \leq i + 34$, то переход к п. 136, иначе к п. 138;

136. аппроксимация $PilotReOKKP[j] = factor_k * j + factor_b;$

137. $j = j + 1$, переход к п. 135;

138. $i = i + 32$, переход к п. 122;

139. $i = 158$;

140. если $i \leq 222$, то переход к п. 141, иначе к п. 157;

141. нахождение коэффициентов для аппроксимации:
 $factor_k = (PilotReOKKP[i+32] - PilotReOKKP[i])/32;$
 $factor_b = PilotReOKKP[i] - factor_k * i;$

142. если $i = 158$, то переход к п. 143, иначе к п. 147;

143. $j = 156;$

144. если $j \leq 157$, то переход к п. 145, иначе к п. 147;

145. аппроксимация для индексов меньше 2
 $PilotReOKKP[j] = factor_k * j + factor_b;$

146. $j = j + 1$, переход к п. 144;

147. $j = i + 1;$

148. если $j \leq i + 31$, то переход к п. 149, иначе к п. 151;

149. аппроксимация $PilotReOKKP[j] = factor_k * j + factor_b;$

150. $j = j + 1$, переход к п. 148;

151. если $i = 222$, то переход к п. 152, иначе к п. 156;

152. $j = i + 33;$

153. если $j \leq i + 33$, то переход к п. 154, иначе к п. 156;

154. аппроксимация $PilotReOKKP[j] = factor_k * j + factor_b;$

155. $j = j + 1$, переход к п. 153;

156. $i = i + 32$, переход к п. 140;

157. повторить п. 121-156 для аппроксимации мнимой части и нахождения массива *PilotImOKKP*.

158. $i = 1;$

159. если $i \leq 100$, то переход к п. 160, иначе к п. 161;

160. комплексное умножение скорректированных реальной и мнимой частей спектра сигнала (*S_RePrin_okkp* и *S_ImPrin_okkp*) на ОККП, определенный по пилот-поднесущим (*PilotReOKKP* и *PilotImOKKP*) для определения реальной и мнимой частей спектра сигнала:
 $S_{RE}[i] = S_RePrin_okkp[i] * PilotReOKKP[i] - S_ImPrin_okkp[i] * PilotImOKKP[i],$
 $S_{IM}[i] = S_ImPrin_okkp[i] * PilotReOKKP[i] + S_RePrin_okkp[i] * PilotImOKKP[i];$

161. $i = 156;$

162. если $i \leq n - 1$, то переход к п. 160, иначе к п. 161;

163. комплексное умножение скорректированных реальной и мнимой частей спектра сигнала (*S_RePrin_okkp* и *S_ImPrin_okkp*) на ОККП, определенный по пилот-поднесущим (*PilotReOKKP* и *PilotImOKKP*) для определения реальной и мнимой частей спектра сигнала:
 $S_{RE}[i] = S_RePrin_okkp[i] * PilotReOKKP[i] - S_ImPrin_okkp[i] * PilotImOKKP[i],$
 $S_{IM}[i] = S_ImPrin_okkp[i] * PilotReOKKP[i] + S_RePrin_okkp[i] * PilotImOKKP[i];$

164. конец.

Используемые функции и процедуры:

Функция ***getModBPSK*** возвращает значения спектральных составляющих (реальная и мнимая часть одного созвездия) из переданного в нее 1 бита входных данных (число от 0 до 1 в десятичной системе счисления).

Функция ***getModQPSK*** возвращает значения спектральных составляющих (реальная и мнимая часть одного созвездия) из переданных в нее 2 бит входных данных (число от 0 до 3 в десятичной системе счисления).

Функция ***getModQAM16*** возвращает значения спектральных составляющих (реальная и мнимая часть одного созвездия) из переданных в нее 4 бит входных данных (число от 0 до 15 в десятичной системе счисления).

Функция ***getModQAM64*** возвращает значения спектральных составляющих (реальная и мнимая часть одного созвездия) из переданных в нее 6 бит входных данных (число от 0 до 63 в десятичной системе счисления).

Процедура ***modulationBPSK*** используя маску, выделяет из исходных данных (1 байт) 1 бит и передает в функцию ***getModBPSK*** для получения спектральных составляющих. Возвращает ***s_BPSK*** - сигнальные созвездия из 8ми комплексных составляющих для кодирования одного байта.

Процедура ***modulationQPSK*** используя маску, выделяет из исходных данных (1 байт) 2 бита и передает в функцию ***getModQPSK*** для получения спектральных составляющих. Возвращает ***s_QPSK*** - сигнальные созвездия из 4х комплексных составляющих для кодирования одного байта.

Процедура ***modulationQAM16*** используя маску, выделяет из исходных данных (1 байт) 4 бита и передает в функцию ***getModQAM16*** для получения спектральных составляющих. Возвращает ***s_QAM16*** - сигнальные созвездия из 2х комплексных составляющих для кодирования одного байта.

Процедура ***modulationQAM64*** используя маску, выделяет из исходных данных (3 байта) 6 бит и передает в функцию ***getModQAM64*** для получения спектральных составляющих. Возвращает ***s_QAM64*** - сигнальные созвездия из 4х комплексных составляющих для кодирования 3х байт.

Процедура ***GetModulationBPSK*** используется для заполнения массивов спектральных составляющих ***OFDM*** символа (реальной и мнимой частей) для модуляции ***BPSK***.

Процедура ***GetModulationQPSK*** используется для заполнения массивов спектральных составляющих *OFDM* символа (реальной и мнимой частей) для модуляции *QPSK*.

Процедура ***GetModulationQAM16*** используется для заполнения массивов спектральных составляющих *OFDM* символа (реальной и мнимой частей) для модуляции *QAM16*.

Процедура ***GetModulationQAM64*** используется для заполнения массивов спектральных составляющих *OFDM* символа (реальной и мнимой частей) для модуляции *QAM64*.

Процедура ***GetDACBits*** преобразует спектр созданного *OFDM* символа (вещественные значения) в значения необходимые для ЦАП в зависимости от битности (8, 10, 12, 14 или 16 бит).

Процедура ***GetDACBitsPreamb*** преобразует спектр преамбулы в значения необходимые для ЦАП в зависимости от битности (8, 10, 12, 14 или 16 бит).

Процедура ***GetNoiseVal*** возвращает по одной шумовой точке из переданных в нее массивов шумовых значений для реальной и мнимой частей сигнала.

Процедура ***FiltrNormalNoise*** фильтрует нормальный шум. В функцию передается тип окна, границы полосы пропускания, массивы шума для реальной и мнимой частей сигнала, возвращает она отфильтрованный в заданной полосе пропускания шум.

Процедура ***CreateWindow*** формирует массив окна Хэмминга или прямоугольного окна. В процедуру передается тип окна, возвращает она сформированный массив окна.

Процедура ***find_max_min*** используется для нахождения максимума и минимума. В процедуру передается массив данных и границы поиска, возвращает она максимальное и минимальное значения.

Процедура ***GetGist*** используется для построения гистограмм.

Функция ***probability*** находит значение функции плотности вероятности для закона распределения Гаусса. Передаем в функцию сигму, возвращает она значение функции плотности вероятности.

Функция ***getNoise*** возвращает значение шума в точке, выбранной случайным образом, и устанавливает, что значение шума в данной точке уже использовано.

Процедура ***getShum*** устанавливает максимум и минимум шума, возвращает массив шумовых значений и реальное количество шумовых точек.

Процедура **GetPreamble** используется для заполнения реальной и мнимой частей преамбулы спектральными компонентами.

Процедура **makePreamble** позволяет получить символ преамбулы во временной области.

Процедура **getOKKP** для определения ОКПХ путем комплексного деления эталонного значения на принятое значение.

Процедура **CreateOneSymbolOFDM** формирует один *OFDM* символ. Сначала посредством вызова функции **GetModulationBPSK** / **GetModulationBPSK** / **GetModulationQAM16** / **GetModulationQAM64** (в зависимости от выбранной модуляции) формируется спектр для 192 активных компонент (8 компонент – пилот-поднесущие), далее первые 100 компонент записываются в массив с 1ой по 100ую, оставшиеся 100 компонент записываются с 156ой по 255ую, компоненты с 101ой по 155ую приравниваются к нулю. Далее полученный спектр переводится во временную область и путем выделения куска сигнала заданной длины в конце и копирования его в начало создается циклический префикс. Затем сформированный *OFDM* символ передается в процедуру **GetDACBits** для получения итогового массива, скорректированного в зависимости от битности ЦАП и типа представления данных.

Функция **getModBPSK** (y : целое число): комплексное число.

y – 1 бит входных данных.

Начало.

1. Если $y=0$, то *Результат.Re*=1, *Результат.Im*=0.

2. Если $y=1$, то *Результат.Re*=1, *Результат.Im*=0.

Конец.

Функция **getModQPSK** (y : целое число): комплексное число.

y – 2 бита входных данных.

Начало.

1. Если $y=0$, то *Результат.Re*=1, *Результат.Im*=1.

2. Если $y=1$, то *Результат.Re*=1, *Результат.Im*=-1.

3. Если $y=2$, то *Результат.Re*=-1, *Результат.Im*=1.

4. Если $y=3$, то *Результат.Re*=-1, *Результат.Im*=-1.

Конец.

Функция **getModQAM16** (y : целое число): комплексное число.

y – 4 бита входных данных.

Начало.

1. Если $y=0$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=1$.
2. Если $y=1$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=3$.
3. Если $y=2$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=1$.
4. Если $y=3$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=3$.
5. Если $y=4$, то $\text{Результат.Re}=-1$, $\text{Результат.Im}=1$.
6. Если $y=5$, то $\text{Результат.Re}=-1$, $\text{Результат.Im}=3$.
7. Если $y=6$, то $\text{Результат.Re}=-3$, $\text{Результат.Im}=1$.
8. Если $y=7$, то $\text{Результат.Re}=-3$, $\text{Результат.Im}=3$.
9. Если $y=8$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=-1$.
10. Если $y=9$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=-3$.
11. Если $y=10$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=-1$.
12. Если $y=11$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=-3$.
13. Если $y=12$, то $\text{Результат.Re}=-1$, $\text{Результат.Im}=-1$.
14. Если $y=13$, то $\text{Результат.Re}=-1$, $\text{Результат.Im}=-3$.
15. Если $y=14$, то $\text{Результат.Re}=-3$, $\text{Результат.Im}=-1$.
16. Если $y=15$, то $\text{Результат.Re}=-3$, $\text{Результат.Im}=-3$.

Конец.

Функция **getModQAM64** (y : целое число): комплексное число.

y – 6 бит ВХОДНЫХ ДАННЫХ.

Начало.

1. Если $y=0$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=3$.
2. Если $y=1$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=1$.
3. Если $y=2$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=5$.
4. Если $y=3$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=7$.
5. Если $y=4$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=-3$.
6. Если $y=5$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=-1$.
7. Если $y=6$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=-5$.
8. Если $y=7$, то $\text{Результат.Re}=3$, $\text{Результат.Im}=-7$.
9. Если $y=8$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=3$.
10. Если $y=9$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=1$.
11. Если $y=10$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=5$.
12. Если $y=11$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=7$.
13. Если $y=12$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=-3$.
14. Если $y=13$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=-1$.
15. Если $y=14$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=-5$.
16. Если $y=15$, то $\text{Результат.Re}=1$, $\text{Результат.Im}=-7$.
17. Если $y=16$, то $\text{Результат.Re}=5$, $\text{Результат.Im}=3$.

18. Если $y=17$, то $Результат.Re=5$, $Результат.Im=1$.
19. Если $y=18$, то $Результат.Re=5$, $Результат.Im=5$.
20. Если $y=19$, то $Результат.Re=5$, $Результат.Im=7$.
21. Если $y=20$, то $Результат.Re=5$, $Результат.Im=-3$.
22. Если $y=21$, то $Результат.Re=5$, $Результат.Im=-1$.
23. Если $y=22$, то $Результат.Re=5$, $Результат.Im=-5$.
24. Если $y=23$, то $Результат.Re=5$, $Результат.Im=-7$.
25. Если $y=24$, то $Результат.Re=1$, $Результат.Im=3$.
26. Если $y=25$, то $Результат.Re=7$, $Результат.Im=1$.
27. Если $y=26$, то $Результат.Re=7$, $Результат.Im=5$.
28. Если $y=27$, то $Результат.Re=7$, $Результат.Im=7$.
29. Если $y=28$, то $Результат.Re=7$, $Результат.Im=-3$.
30. Если $y=29$, то $Результат.Re=7$, $Результат.Im=-1$.
31. Если $y=30$, то $Результат.Re=7$, $Результат.Im=-5$.
32. Если $y=31$, то $Результат.Re=7$, $Результат.Im=-7$.
33. Если $y=32$, то $Результат.Re=-3$, $Результат.Im=3$.
34. Если $y=33$, то $Результат.Re=-3$, $Результат.Im=1$.
35. Если $y=34$, то $Результат.Re=-3$, $Результат.Im=5$.
36. Если $y=35$, то $Результат.Re=-3$, $Результат.Im=7$.
37. Если $y=36$, то $Результат.Re=-3$, $Результат.Im=-3$.
38. Если $y=37$, то $Результат.Re=-3$, $Результат.Im=-1$.
39. Если $y=38$, то $Результат.Re=-3$, $Результат.Im=-5$.
40. Если $y=39$, то $Результат.Re=-3$, $Результат.Im=-7$.
41. Если $y=40$, то $Результат.Re=-1$, $Результат.Im=3$.
42. Если $y=41$, то $Результат.Re=-1$, $Результат.Im=1$.
43. Если $y=42$, то $Результат.Re=-1$, $Результат.Im=5$.
44. Если $y=43$, то $Результат.Re=-1$, $Результат.Im=7$.
45. Если $y=44$, то $Результат.Re=-1$, $Результат.Im=-3$.
46. Если $y=45$, то $Результат.Re=-1$, $Результат.Im=-1$.
47. Если $y=46$, то $Результат.Re=-1$, $Результат.Im=-5$.
48. Если $y=47$, то $Результат.Re=-1$, $Результат.Im=-7$.
49. Если $y=48$, то $Результат.Re=-5$, $Результат.Im=3$.
50. Если $y=49$, то $Результат.Re=-5$, $Результат.Im=1$.
51. Если $y=50$, то $Результат.Re=-5$, $Результат.Im=5$.
52. Если $y=51$, то $Результат.Re=-5$, $Результат.Im=7$.
53. Если $y=52$, то $Результат.Re=-5$, $Результат.Im=-3$.
54. Если $y=53$, то $Результат.Re=-5$, $Результат.Im=-1$.
55. Если $y=54$, то $Результат.Re=-5$, $Результат.Im=-5$.

56. Если $y=55$, то $Результат.Re=-5$, $Результат.Im=-7$.
 57. Если $y=56$, то $Результат.Re=-1$, $Результат.Im=3$.
 58. Если $y=57$, то $Результат.Re=-7$, $Результат.Im=1$.
 59. Если $y=58$, то $Результат.Re=-7$, $Результат.Im=5$.
 60. Если $y=59$, то $Результат.Re=-7$, $Результат.Im=7$.
 61. Если $y=60$, то $Результат.Re=-7$, $Результат.Im=-3$.
 62. Если $y=61$, то $Результат.Re=-7$, $Результат.Im=-1$.
 63. Если $y=62$, то $Результат.Re=-7$, $Результат.Im=-5$.
 64. Если $y=63$, то $Результат.Re=-7$, $Результат.Im=-7$.
- Конец.

Процедура **modulationBPSK** (входной параметр x : целое число, выходной параметр s : тип m_BPSK (тип данных m_BPSK массив из 8ми элементов $CPLX$, тип данных $CPLX$ - запись содержащая поля Re и Im)).

x – исходные данные,

s - сигнальные созвездия из 8ми комплексных составляющих для кодирования одного байта.

Переменные b, i, y, n_steps : целые числа;

Начало

1. $n_steps=8$ (требуется 8 шагов для передачи 1 байта, т.к. в $BPSK$ 1 созвездие несет 1 бит данных);
2. $i=0$;
3. если $i < n_steps-1$, то переход к п. 4, иначе переход к п. 10;
4. $b=1 \text{ shl } (1*i)$, b – маска для выделения из числа 1 бита. Например, $i=0$, $b=00000001$; $i=1$, $b=00000010$; $i=7$, $b=10000000$; $\text{shl } 1$ – сдвиг влево на 1 символ;
5. $y=x \text{ and } b$, применяем маску к числу x и выделяем из него 1 бит;
6. $y=y \text{ shr } (1*i)$, сдвигаем вправо полученное число для получения символа от 0 до 1, shr – сдвиг вправо;
7. $s[i]=getModBPSK(y)$, передаем число y в функцию $getModBPSK$ для получения массива s спектральных составляющих;
8. получение массива s ;
9. $i=i+1$, переход к п. 3;
10. конец.

Процедура **modulationQPSK** (входной параметр x : целое число, выходной параметр s : тип m_QPSK (тип данных m_QPSK массив из 4 элементов $CPLX$, тип данных $CPLX$ - запись содержащая поля Re и Im)).

x – исходные данные,
 s - сигнальные созвездия из 4х комплексных составляющих для кодирования одного байта.

Переменные b, i, y, n_steps : целые числа;

Начало

1. $n_steps=4$ (требуется 4 шага для передачи 1 байта, т.к. в QPSK 1 созвездие несет 2 бита данных);
2. $i=0$;
3. если $i < n_steps-1$, то переход к п. 4, иначе переход к п. 10;
4. $b=3 \text{ shl } (2*i)$, b – маска для выделения из числа 2 бита (3 в двоичной системе 11). Например, $i=0, b=00000011$; $i=1, b=00001100$; $i=3, b=11000000$; $\text{shl } 2$ – сдвиг влево на 2 символа;
5. $y=x \text{ and } b$, применяем маску к числу x и выделяем из него 2 бита;
6. $y=y \text{ shr } (2*i)$, сдвигаем вправо на 2 символа для получения числа от 0 до 3, shr – сдвиг вправо;
7. $s[i]=\text{getModBPSK}(y)$, передаем число y в функцию getModQPSK для получения массива с спектральными составляющими;
8. получение массива s ;
9. $i=i+1$, переход к п. 3;
10. конец.

Процедура **modulationQAM16** (входной параметр x : целое число, выходной параметр s : тип m_QAM16 (тип данных m_QAM16 массив из 2 элементов $CPLX$, тип данных $CPLX$ - запись содержащая поля Re и Im)).

x – исходные данные,

s - сигнальные созвездия из 2х комплексных составляющих для кодирования одного байта.

Переменные b, i, y, n_steps : целые числа;

Начало

1. $n_steps=2$ (требуется 2 шага для передачи 1 байта, т.к. в QAM16 1 созвездие несет 4 бита данных);
2. $i=0$;
3. если $i < n_steps-1$, то переход к п. 4, иначе переход к п. 10;
4. $b=15 \text{ shl } (4*i)$, b – маска для выделения из числа 4 бита (15 в двоичной системе 1111). Например, $i=0, b=00001111$; $i=1, b=11110000$; $\text{shl } 4$ – сдвиг влево на 4 символа;
5. $y=x \text{ and } b$, применяем маску к числу x и выделяем из него 4 бита;

6. $y = y \text{ shr } (4*i)$, смещаем вправо на 4 символа для получения числа от 0 до 15, shr – сдвиг вправо;
7. $s[i] = \text{getModBPSK}(y)$, передаем число y в функцию getModQAM16 для получения массива s спектральных составляющих;
8. получение массива s ;
9. $i = i + 1$, переход к п. 3;
10. конец.

Процедура ***modulationQAM64*** (входные параметры $x0$, $x1$, $x2$: целые числа, выходной параметр s : тип m_QAM64 (тип данных m_QAM64 массив из 4 элементов $CPLX$, тип данных $CPLX$ - запись содержащая поля Re и Im)).

x – исходные данные,

s - сигнальные созвездия из 4х комплексных составляющих для кодирования 3х байт.

Переменная $v6$: целое число;

Начало

1. т.к. 1 сигнальное созвездие $QAM64$ передает 6 бит, то 3 числа, состоящие из 8 бит ($3*8=24$) разделяем на 4 куса по 6 бит.
 $v6 = x0 \text{ and } 63$, из числа $x0$ выделяем первые 6 бит (63 в двоичной системе 111111);
2. $s[0] = \text{getModQAM64}(v6)$, передаем $v6$ в функцию getModQAM64 для получения спектральных составляющих;
3. $v6 = x0 \text{ and } 192$, из числа $x0$ выделяем последние 2 бита (192 в двоичной системе 11000000);
4. $v6 = v6 \text{ shr } 6$, смещаем полученные 2 бита вправо на 6 позиций и они становятся 2 первыми битами;
5. $v6 = v6 + ((x1 \text{ and } 15) \text{ shl } 2)$, добавляем к этим 2 битам первые 4 бита числа $x1$, смещенные на 2 позиции;
6. $s[1] = \text{getModQAM64}(v6)$, передаем $v6$ (последние 2 бита числа $x0$, которые стали 2мя первыми и первые 4 бита числа $x1$, которые стали 3им, 4ым, 5ым и 6ым) в функцию getModQAM64 для получения спектральных составляющих;
7. $v6 = x1 \text{ and } 240$, из числа $x1$ выделяем оставшиеся 4 бита (240 в двоичной системе 11110000);
8. $v6 = v6 \text{ shr } 4$, смещаем полученные 4 бита вправо на 4 позиции и они становятся 4мя первыми;

9. $v6 = v6 + ((x2 \text{ and } 3) \text{ shl } 4)$, добавляем к этим 4-м битам первые 2 бита числа $x2$, смещенные на 4 позиции, при этом они становятся 5-м и 6-м битами;
10. $s[2] = \text{getModQAM64}(v6)$, передаем число $v6$ (последние 4 бита числа $x1$, которые стали первыми и первые 2 бита числа $x2$, которые стали 5-м и 6-м битами) в функцию getModQAM64 для получения спектральных составляющих;
11. $v6 = x2 \text{ and } 252$, из числа $x2$ выделяем оставшиеся 6 бит (252 в двоичной системе 11111100);
12. $s[3] = \text{getModQAM64}(v6)$, передаем число $v6$ (последние 6 бит числа $x2$) в функцию getModQAM64 для получения спектральных составляющих;
13. конец.

Процедура **GetModulationBPSK** (входные параметры *in_data*: массив целых чисел, *n_byte_ofdm*: целое число, выходные параметры *sRe*, *sIm*: массивы целых чисел).

in_data – исходные данные,

n_byte_ofdm - кол-во байт в OFDM символе,

sRe, *sIm* – массивы спектральных составляющих OFDM символа (реальная и мнимая части).

Переменные *i*, *j*, *k*, *n_steps*: целые числа; *s_BPSK*: *m_BPSK* (тип *m_BPSK* описан выше).

Начало

1. $n_steps = 8, j = 0, i = 0$;
2. если $i < n_byte_ofdm - 1$, то переход к п. 3, иначе к п. 8;
3. вызов процедуры *modulationBPSK*, в которую передаем входные данные;
4. $k = 0$;
5. если $k < n_steps - 1$, тогда переход к п. 6, иначе к п. 7;
6. заполняем массив спектральными составляющими
 $sRe[j] = s_BPSK[k].Re,$
 $sIm[j] = s_BPSK[k].Im,$
 $j = j + 1,$
 $k = k + 1,$
переход к п. 5;
7. $i = i + 1$, переход к п. 2;
8. конец.

Процедура **GetModulationQPSK** (входные данные *in_data*: массив целых чисел, *n_byte_ofdm*: целое число, выходные данные *sRe*, *sIm*: массивы целых чисел).

in_data – исходные данные,

n_byte_ofdm -кол-во байт в *OFDM* символе,

sRe, *sIm* – массивы спектральных составляющих *OFDM* символа (реальная и мнимая части).

Переменные *i*, *j*, *k*, *n_steps*: целые числа, *s_QPSK*: *m_QPSK* (тип *m_QPSK* описан выше).

Начало

1. *n_steps*=4, *j*=0, *i*=0;
2. если *i* < *n_byte_ofdm*-1, то переход к п. 3, иначе к п. 8;
3. вызов процедуры *modulationQPSK*, в которую передаем входные данные;
4. *k*=0;
5. если *k* < *n_steps*-1, тогда переход к п. 6, иначе к п. 7;
6. заполняем массив спектральными составляющими
sRe[*j*]=*s_QPSK*[*k*].*Re*,
sIm[*j*]=*s_QPSK*[*k*].*Im*,
j=*j*+1,
k=*k*+1,
переход к п. 5;
7. *i*=*i*+1, переход к п. 2;
8. конец.

Процедура **GetModulationQAM16** (входные данные *in_data*: массив целых чисел, *n_byte_ofdm*: целое число, выходные данные *sRe*, *sIm*: массивы целых чисел).

in_data – исходные данные,

n_byte_ofdm -кол-во байт в *OFDM* символе,

sRe, *sIm* – массивы спектральных составляющих *OFDM* символа (реальная и мнимая части).

Переменные *i*, *j*, *k*, *n_steps*: целые числа, *s_QAM16*: *m_QAM16* (тип *m_QAM16* описан выше).

Начало

1. *n_steps*=2, *j*=0, *i*=0;
2. если *i* < *n_byte_ofdm*-1, то переход к п. 3, иначе к п. 8;

3. вызов процедуры *modulationQAM16*, в которую передаем входные данные;
4. $k=0$;
5. если $k < n_steps-1$, тогда переход к п. 6, иначе к п. 7;
6. заполняем массив спектральными составляющими
 $sRe[j]=s_QAM16[k].Re$,
 $sIm[j]=s_QAM16[k].Im$,
 $j=j+1$,
 $k=k+1$,
переход к п. 5;
7. $i=i+1$, переход к п. 2;
8. конец.

Процедура ***GetModulationQAM64*** (входные данные *in_data[i]*: массив целых чисел, *n_byte_ofdm*: целое число, выходные данные *sRe*, *sIm*: массивы целых чисел).

in_data – исходные данные,

n_byte_ofdm -кол-во байт в *OFDM* символе,

sRe, *sIm* – массивы спектральных составляющих *OFDM* символа (реальная и мнимая части).

Переменные *i*, *j*, *k*, *n_steps*: целые числа, *s_QAM64*: *m_QAM64* (тип *m_QAM64* описан выше).

Начало

1. $n_steps=4$, $j=0$, $i=0$;
2. если $i < n_byte_ofdm-1$, то переход к п. 3, иначе к п. 8;
3. вызов процедуры *modulationQAM64*, в которую передаем входные данные;
4. $k=0$;
5. если $k < n_steps-1$, тогда переход к п. 6, иначе к п. 7;
6. заполняем массив спектральными составляющими
 $sRe[j]=s_QAM64[k].Re$,
 $sIm[j]=s_QAM64[k].Im$,
 $j=j+1$,
 $k=k+1$,
переход к п. 5;
7. $i=i+3$, переход к п. 2;
8. конец.

Процедура **GetDACBits** (входные параметры i : целое число; a, b_bit : вещественные числа; входные и выходные параметры $sRe_ofdm_itog, sIm_ofdm_itog$: массивы вещественных чисел);

i – индекс,

a, b_bit - корректирующие множители в зависимости от битности ЦАП,

sRe_ofdm_itog - OFDM символ с циклическим префиксом, реальная часть,

sIm_ofdm_itog - OFDM символ с циклическим префиксом, мнимая часть.

Начало

1. если $i=2$ или $i=34$ или $i=66$ или $i=98$ или $i=158$ или $i=190$ или $i=222$ или $i=254$, тогда выход, иначе переход к п. 2;

2. если dac_bits8 , тогда переход к п. 3, иначе к п. 4;

3. $sRe_ofdm_itog[i]=sRe_ofdm_itog[i]*b_bit*a$;
 $sIm_ofdm_itog[i]=sIm_ofdm_itog[i]*b_bit*a$.

Домножаем итоговый массив выходных данных на соответствующий коэффициент так, чтобы для 8битного ЦАП сигнал лежал в пределах от -128 до 127;

4. если dac_bits10 , тогда переход к п. 5, иначе к п. 6;

5. $sRe_ofdm_itog[i]=sRe_ofdm_itog[i]*b_bit*a$;
 $sIm_ofdm_itog[i]=sIm_ofdm_itog[i]*b_bit*a$.

Домножаем итоговый массив выходных данных на соответствующий коэффициент так, чтобы для 10битного ЦАП сигнал лежал в пределах от -512 до 511;

6. если dac_bits12 , тогда переход к п. 7, иначе к п. 8;

7. $sRe_ofdm_itog[i]=sRe_ofdm_itog[i]*b_bit*a$;
 $sIm_ofdm_itog[i]=sIm_ofdm_itog[i]*b_bit*a$.

Домножаем итоговый массив выходных данных на соответствующий коэффициент так, чтобы для 12битного ЦАП сигнал лежал в пределах от -2048 до 2047;

8. если dac_bits14 , тогда переход к п. 9, иначе к п. 10;

9. $sRe_ofdm_itog[i]=sRe_ofdm_itog[i]*b_bit*a$;
 $sIm_ofdm_itog[i]=sIm_ofdm_itog[i]*b_bit*a$.

Домножаем итоговый массив выходных данных на соответствующий коэффициент так, чтобы для 14битного ЦАП сигнал лежал в пределах от -8192 до 8191;

10. если dac_bits16 , тогда переход к п. 11, иначе к п. 12;

11. $sRe_ofdm_itog[i]=sRe_ofdm_itog[i]*b_bit*a$;

$$sIm_ofdm_itog[i] = sIm_ofdm_itog[i] * b_bit * a.$$

Домножаем итоговый массив выходных данных на соответствующий коэффициент так, чтобы для 16битного ЦАП сигнал лежал в пределах от -32768 до 32767;

12. конец.

Процедура **GetDACBitspreamb** (входные параметры i : целое число; a , b_bit : вещественные числа; входные и выходные параметры sRe_ofdm_itog , sIm_ofdm_itog : массивы вещественных чисел);

i – индекс,

a , b_bit - корректирующие множители в зависимости от битности ЦАП,

sRe_ofdm_itog - OFDM символ с циклическим префиксом, реальная часть,

sIm_ofdm_itog - OFDM символ с циклическим префиксом, мнимая часть.

Начало

1. если dac_bits8 , тогда переход к п. 2, иначе к п. 3;

2. $sRe_ofdm_itog[i] = sRe_ofdm_itog[i] * b_bit * a$;
 $sIm_ofdm_itog[i] = sIm_ofdm_itog[i] * b_bit * a$.

Домножаем итоговый массив выходных данных на соответствующий коэффициент так, чтобы для 8битного ЦАП сигнал лежал в пределах от -128 до 127;

3. если dac_bits10 , тогда переход к п. 4, иначе к п. 5;

4. $sRe_ofdm_itog[i] = sRe_ofdm_itog[i] * b_bit * a$;
 $sIm_ofdm_itog[i] = sIm_ofdm_itog[i] * b_bit * a$.

Домножаем итоговый массив выходных данных на соответствующий коэффициент так, чтобы для 10битного ЦАП сигнал лежал в пределах от -512 до 511;

5. если dac_bits12 , тогда переход к п. 6, иначе к п. 7;

6. $sRe_ofdm_itog[i] = sRe_ofdm_itog[i] * b_bit * a$;
 $sIm_ofdm_itog[i] = sIm_ofdm_itog[i] * b_bit * a$.

Домножаем итоговый массив выходных данных на соответствующий коэффициент так, чтобы для 12битного ЦАП сигнал лежал в пределах от -2048 до 2047;

7. если dac_bits14 , тогда переход к п. 8, иначе к п. 9;

8. $sRe_ofdm_itog[i] = sRe_ofdm_itog[i] * b_bit * a$;
 $sIm_ofdm_itog[i] = sIm_ofdm_itog[i] * b_bit * a$.

Домножаем итоговый массив выходных данных на соответствующий коэффициент так, чтобы для 14битного ЦАП сигнал лежал в пределах от -8192 до 8191;

9. если *dac_bits*16, тогда переход к п. 10, иначе к п. 11;

10. $sRe_ofdm_itog[i] = sRe_ofdm_itog[i] * b_bit * a;$
 $sIm_ofdm_itog[i] = sIm_ofdm_itog[i] * b_bit * a.$

Домножаем итоговый массив выходных данных на соответствующий коэффициент так, чтобы для 16битного ЦАП сигнал лежал в пределах от -32768 до 32767;

11. конец.

Процедура **GetNoiseVal** (входные параметры *noiseIndexMax*, *noiseIndexMin*: целые числа; *shumRe*, *shumIm*: массивы вещественных чисел; выходные параметры *shumReOne*, *shumImOne*: вещественные числа).

noiseIndexMax - верхняя граница шума,

noiseIndexMin - нижняя граница шума,

shumRe - шум для реальной части сигнала,

shumIm - шум для мнимой части сигнала,

shumReOne - одна точка шума для реальной части сигнала,

shumImOne - одна точка шума для мнимой части сигнала.

Начало

1. если *noiseIndex*>*noiseIndexMax*, тогда переход к п. 2, иначе переход к п. 3;

2. *noiseIndex*=*noiseIndexMin*;

3. *shumReOne*=*shumRe*[*noiseIndex*];

shumImOne=*shumIm*[*noiseIndex*];

noiseIndex=*noiseIndex*+1;

4. конец.

Процедура **FiltrNormalNoise** (входные параметры *window_type*, *kol_tochek_signal*, *f_disk*, *f1*, *f2*: целое число; выходные параметры *shumRe*, *shumIm*: массивы вещественных чисел).

f1 - нижняя граница полосы пропускания,

f2 - верхняя граница полосы пропускания,

kol_tochek_signal - кол-во точек для построения сигнала,

f_disk - частота дискретизации,

window_type – переменная для определения используется окно для АЧХ или нет,

shumRe - шум для реальной части сигнала,

shumIm - шум для мнимой части сигнала.

Переменные *i, j*: целые числа;

x_reACHH, *x_imACHH*, *filterRe*, *filterIm*, *windowACHH*: массивы вещественных чисел.

Начало

1. $i=0$;
2. если $i < kol_tochek_signal-1$, тогда переход к п. 3, иначе к п. 5;
3. $x_reACHH[i]=0$; $x_imACHH[i]=0$, нулим все компоненты;
4. $i=i+1$, переход к п. 2;
5. $i = (f1/f_disk)*kol_tochek_signal$;
6. если $i < (f2/f_disk)*kol_tochek_signal$, тогда переход к п. 7, иначе к п. 9;
7. $x_reACHH[i]=1$, для полосы пропускания от $f1$ до $f2$ приравниваем компоненты к 1;
8. $i=i+1$, переход к п. 6;
9. $j=0$;
10. $i = kol_tochek_signal-1$;
11. если $i > kol_tochek_signal/2$, тогда переход к п. 12, иначе к п. 14;
12. $j=j+1$;
 $x_reACHH[i]=x_reACHH[j]$;
 $x_imACHH[i]=x_imACHH[j]$;
13. $i=i-1$, переход к п. 11;
14. вызов процедуры *FFT*, обратное преобразование Фурье для получения *x_reACHH*, *x_imACHH* во временной области;
15. Вызов процедуры *CreateWindow*, создаем массив окна (*windowACHH*);
16. $i=0$;
17. если $i < kol_tochek_signal/2-1$, тогда переход к п. 18, иначе к п. 20;
18. $x_reACHH[i]=x_reACHH[i]*windowACHH[i + kol_tochek_signal/2]$;
 $x_imACHH[i]=x_imACHH[i]*windowACHH[i + kol_tochek_signal/2]$,
умножаем АЧХ на окно;
19. $i=i+1$, переход к п. 17;
20. $i = kol_tochek_signal/2$;
21. если $i < kol_tochek_signal-1$, тогда переход к п. 22, иначе к п. 24;
22. $x_reACHH[i]=x_reACHH[i]*windowACHH[i - kol_tochek_signal/2]$;
 $x_imACHH[i]=x_imACHH[i]*windowACHH[i - kol_tochek_signal/2]$,
умножаем вторую часть АЧХ на окно;

23. $i=i+1$, переход к п. 21;
24. вызов процедуры *FFT*, прямое преобразование Фурье для получения АЧХ с окном x_{reACHH} , x_{imACHH} в частотной области;
25. вызов процедуры *FFT*, прямое преобразование Фурье для получения шума $shumRe$, $shumIm$ в частотной области;
26. $filterRe = x_{reACHH}$;
 $filterIm = x_{imACHH}$;
27. $i=0$;
28. если $i < kol_tochek_signal-1$, тогда переход к п. 29, иначе к п. 31;
29. $shumRe[i] = shumRe[i] * filterRe[i] - shumIm[i] * filterIm[i]$;
 $shumIm[i] = filterRe[i] * shumIm[i] + shumRe[i] * filterIm[i]$,
умножаем реальную и мнимую части шума на фильтрованную АЧХ для получения фильтрованного шума $shumRe$ и $shumIm$;
30. $i=i+1$, переход к п. 28;
31. вызов процедуры *FFT*, обратное преобразование Фурье для получения фильтрованных реальной ($shumRe$) и мнимой ($shumIm$) частей шума во временной области;
32. конец.

Процедура **CreateWindow** (входной параметр *window_type*: целое число; выходной параметр *windowACHH*: массив вещественных чисел);

window_type – переменная для определения используется окно или нет,
windowACHH – массив окна.

Переменные i , n : целые числа; a : вещественное число.

Начало

1. $a=0.54$;
2. n = размер массива *windowACHH*;
3. если *window_type*=1, тогда переход к п. 4, иначе переход к п. 8;
4. $i=0$;
5. если $i < n-1$, тогда переход к п. 6, иначе к п. 8;
6. $windowACHH[i]=1$;
7. $i=i+1$, переход к п. 5;
8. если *window_type*=2, тогда переход к п. 9, иначе переход к п. 13;
9. $i=0$;
10. если $i < n-1$, тогда переход к п. 11, иначе к п. 13;
11. $windowACHH[i] = a + (1-a) * \cos(2 * \pi * i / n + \pi)$;
12. $i=i+1$, переход к п. 10;
13. конец.

Функция **probability** (*sigma*, *z*: вещественные числа): вещественное число;

sigma - сигма,

z - переменная для построения закона распределения шума,

Начало

$$1. \text{ Result} = \frac{1}{\sqrt{2\pi} \text{ sigma}} e^{\frac{-z^2}{2\text{sigma}^2}};$$

2. конец.

Функция **getNoise** (*shum*: массив типа *noise_t*; *N*: целое число): вещественное число;

N – количество шумовых точек,

shum - массив шумовых значений.

Переменная *i*: целое число;

Начало

1. *i*=*Random*(*N*), случайным образом выбираем индекс *i* из чисел от 0 до *N*;

2. если *not shum[i].isUsed*, т.е. индекс еще не использовался, тогда переход к п. 3, иначе к п. 4;

3. *Result*=*shum[i].noise*, возвращаем значение шума в *i*той точке;
shum[i].isUsed=*True*, устанавливаем, что данный индекс использован;

4. конец.

Процедура **getShum** (*kol_tochek*, *granica2*, *intervalsCount*: целые числа; *sigma*: вещественное число; *noise*: массив типа *noise_t*; выходные параметры *shum*: массив вещественных чисел; *j*: целое число);

kol_tochek - кол-во точек для построения шума,

granica2= (*n*+*n_cp*)* *ofdm_symbols_count*-1,

intervalsCount – количество интервалов,

sigma – сигма,

noise – массив шума,

shum - массив шумовых значений,

j - количество точек, попавших во все промежутки.

Переменные *i*, *N*: целые числа; *z*, *x1*, *x2*, *step*: вещественные числа;

Начало

1. $x1 = -3 * \sigma$;
 $x2 = 3 * \sigma$;
2. $step = (x2 - x1) / intervalsCount$;
 $z = x1$;
 $j = 0$;
3. если $z < x2$, тогда переход к п. 4, иначе к п. 10;
4. вызов функции *probability* для получения закона распределения шума (*trunc*-отброс дробной части числа без округления),
 $N = trunc(probability * step * kol_tochek)$;
5. $i = 1$;
6. если $i < N$, тогда переход к п. 7, иначе к п. 9;
7. $noise[j].noise = z$, для каждой точки записываем свое значение z от $x1$ до $x2$ с шагом $step$;
 $noise[j].isUsed = false$, устанавливаем, что индексы от 0 до $kol_tochek - 1$ еще не использованы;
 $j = j + 1$, определяем количество точек, попавших во все промежутки;
8. $i = i + 1$, переход к п. 6;
9. $z = z + step$, переход к п. 3;
10. $i = 0$;
11. если $i < j - 1$, тогда переход к п. 12, иначе к п. 14;
12. вызов функции *getNoise* для заполнения массива шума *shum*;
13. $i = i + 1$, переход к п. 11;
14. конец.

Процедура **CreateOneSymbolOFDM** (входные параметры *in_data*: массив целых чисел; *preambRe*, *preambIm*, *shumRe*, *shumIm*: массивы вещественных чисел; *ind*, *n_byte_ofdm*, *ofdm_symbols_count*, *n*, *n_cp*, *n_DAC_Bits*, *n_komponent*, *noiseIndexMax*, *noiseIndexMin*, *mod_type*: целые числа; *b_bit*: вещественное число; выходные параметры *specRe*, *specIm*, *signalReItog*, *signalImItog*, *PilotReIdeal*, *PilotImIdeal*: массивы вещественных чисел);

in_data - входные данные,

ind – индекс,

mod_type - определяет вид модуляции,

n_komponent - кол-во активных компонент в *OFDM* символе,

n_byte_ofdm - (кол-во байт в *OFDM* символе,

ofdm_symbols_count - кол-во *OFDM* символов для генерации,

n_DAC_Bits - битность ЦАП,

n - размерность FFT ,

n_{cp} - длина циклического префикса $OFDM$ символа,

b_{bit} - корректирующий множитель в зависимости от битности ЦАП,

$signalReItog$ – выходной массив $OFDM$ символа, реальная часть,

$signalImItog$ – выходной массив $OFDM$ символа, мнимая часть,

$specRe$ - реальная часть спектра,

$specIm$ - мнимая часть спектра,

Переменные:

$shumReOne$, $shumImOne$, a : вещественные числа;

i , k : целые числа;

sRe , sIm : массивы целых чисел;

sRe_ofdm , sIm_ofdm , $cpRe$, $cpIm$, sRe_ofdm_itog , sIm_ofdm_itog : массивы вещественных чисел;

Начало

1. если $mod_type=cQAM64$, тогда вызов процедуры $GetModulationQAM64$ для получения массивов спектральных составляющих sRe , sIm , $a=0,075$;
2. иначе, если $mod_type=cBPSK$, тогда вызов процедуры $GetModulationBPSK$ для получения массивов спектральных составляющих sRe , sIm , $a=0,5$;
3. иначе, если $mod_type=cQPSK$, тогда вызов процедуры $GetModulationQPSK$ для получения массивов спектральных составляющих sRe , sIm , $a=0,37$;
4. иначе, если $mod_type=cQAM16$, тогда вызов процедуры $GetModulationQAM16$ для получения массивов спектральных составляющих sRe , sIm , $a=0,17$;
5. $i=0$;
6. если $i < n-1$, тогда переход к п. 7, иначе к п. 9;
7. $sRe_ofdm[i] = 0$; $sIm_ofdm[i] = 0$, приравниваем к нулю все компоненты, чтобы получить нули в промежутке от 101 до 155;
8. $i=i+1$, переход к п. 6;
9. $i=1$, $k=0$;
10. если $i < 100$, тогда переход к п. 11, иначе к п. 15;
11. если $i=2$ или $i=34$ или $i=66$ или $i=98$, тогда переход к п. 12, иначе к п. 13;
12. добавление пилот-поднесущих:
 $sRe_ofdm[i] = b_{bit} * preambRe[i]$;
 $sIm_ofdm[i] = b_{bit} * preambIm[i]$;

$PilotReIdeal[i] = preambRe[i]/a;$
 $PilotImIdeal[i] = preambIm[i]/a;$
 13. $sRe_ofdm[i] = sRe[k]; sIm_ofdm[i] = sIm[k]$, с 1 по 100 записываем компоненты полученные выше из процедуры *GetModulationBPSK/QPSK/QAM16/QAM64*;
 14. $i=i+1, k=k+1$, переход к п. 10;
 15. $i = 156, k=96$;
 16. если $i < n-1$, тогда переход к п. 17, иначе к п. 21;
 17. если $i=158$ или $i=190$ или $i=222$ или $i=254$, тогда переход к п. 18, иначе к п. 19;
 18. добавление пилот-поднесущих:
 $sRe_ofdm[i] = b_bit * preambRe[i];$
 $sIm_ofdm[i] = b_bit * preambIm[i];$
 $PilotReIdeal[i] = preambRe[i]/a;$
 $PilotImIdeal[i] = preambIm[i]/a;$
 19. $sRe_ofdm[i] = sRe[k];$
 $sIm_ofdm[i] = sIm[k]$, со 156 по $n-1$ записываем компоненты полученные выше из процедуры *GetModulationBPSK/QPSK/QAM16/QAM64*;
 20. $i=i+1, k=k+1$, переход к п. 16;
 21. $i=0$;
 22. если $i < n-1$, тогда переход к п. 23, иначе к п. 27;
 23. $specRe[i] = sRe_ofdm[i];$
 $specIm[i] = sIm_ofdm[i]$, получаем реальную (*specRe*) и мнимую (*specIm*) части спектра для *OFDM* символа;
 24. если $i=2$ или $i=34$ или $i=66$ или $i=98$ или $i=158$ или $i=190$ или $i=222$ или $i=254$, тогда переход к п. 25, иначе к п. 27;
 25. $specRe[i] = sRe_ofdm[i];$
 $specIm[i] = sIm_ofdm[i]$, получаем реальную (*specRe*) и мнимую (*specIm*) части спектра для пилот-поднесущих;
 26. $i=i+1$, переход к п. 22;
 27. $i=0$;
 28. если $i < n-1$, тогда переход к п. 29, иначе к п. 31;
 29. приводим данные к определенным значениям в соответствии с битностью ЦАП. Путем запуска процедуры *GetDACBits* получаем скорректированные массивы sRe_ofdm, sIm_ofdm ;
 30. $i=i+1$, переход к п. 28;

31. *FFT*, берем обратное преобразование Фурье, переводим во временную область массивы *sRe_ofdm*, *sIm_ofdm*;

32. $i=0$;

33. если $i < n_{cp}-1$, тогда переход к п. 34, иначе к п. 36;

34. $sRe_ofdm_itog[i] = sRe_ofdm[i+n-n_{cp}]$;
 $sIm_ofdm_itog[i] = sIm_ofdm[i+n-n_{cp}]$, создаем циклический префикс путем выделения куска сигнала длиной n_{cp} в конце *OFDM* символа и копирования его в начало символа;

35. $i=i+1$, переход к п. 33;

36. $i = n_{cp}$;

37. если $i < n+n_{cp}-1$, тогда переход к п. 38, иначе к п. 40;

38. $sRe_ofdm_itog[i] = sRe_ofdm[i-n_{cp}]$;
 $sIm_ofdm_itog[i] = sIm_ofdm[i-n_{cp}]$, после префикса заполняем массив данными сигнала;

39. $i=i+1$, переход к п. 37;

40. $i=0$;

41. если $i < n+n_{cp}-1$, тогда переход к п. 42, иначе к п. 44;

42. $sRe_ofdm_itog[i] = round(sRe_ofdm_itog[i]*n/2)$;
 $sIm_ofdm_itog[i] = round(sIm_ofdm_itog[i]*n/2)$, округляем массивы;

43. $i=i+1$, переход к п. 41;

44. $i=0$;

45. если $i < n+n_{cp}-1$, тогда переход к п. 46, иначе к п. 49;

46. вызов процедуры *GetNoiseVal* для получения одной точки шума для реальной и мнимой частей *shumReOne*, *shumImOne*;

47. Получение итоговых массивов сигнала совместно с шумом, путем сложения сигнала с шумом:
 $signalReItog[i] = sRe_ofdm_itog[i] + shumReOne$,
 $signalImItog[i] = sIm_ofdm_itog[i] + shumImOne$;

48. $i=i+1$, переход к п. 45;

49. конец.

П.2. Моделирование модифицированного алгоритма коррекции КПХ канала связи с уменьшенным СКЗ

Представленные ниже гистограммы получены с учетом уменьшения в 1,1007 раз СКЗ преамбулы. Такое уменьшение позволяет сделать пиковое значение преамбулы таким же, как и в случае стандартной преамбулы, в которой частотные компоненты идут через одну составляющую. Уменьшение СКЗ преамбулы приводит к тому, что отношение сигнал-шум для преамбулы будет хуже на $20\log(1.1007) \approx 0.83$ дБ. Однако отношение сигнал/шум для символов данных не изменится.

На рисунках 2.1 - 2.7 представлены гистограммы распределений максимального уровня принятого сигнала, где b – битность ЦАП, SNR – отношение сигнал/шум, σ – СКО, d – доверительный интервал.

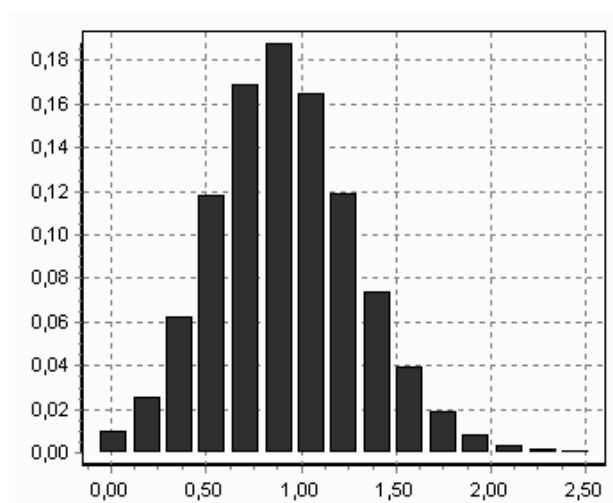


Рисунок 2.1 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=17$ дБ, $\sigma=0,3777$, $d(0,38;1,62)$)

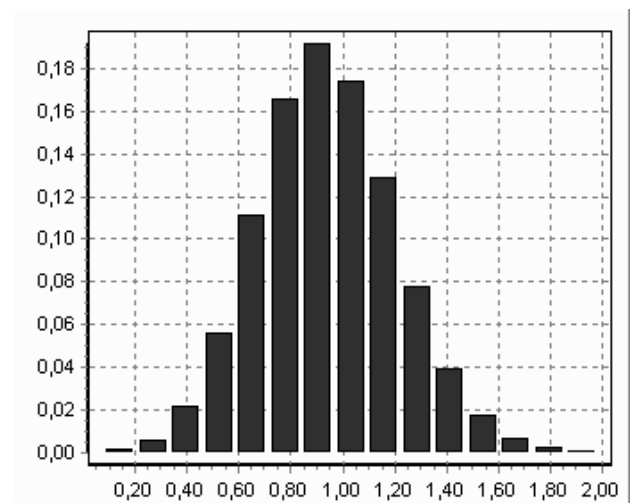


Рисунок 2.2 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=18$ дБ, $\sigma=0,3364$, $d(0,45;1,55)$)

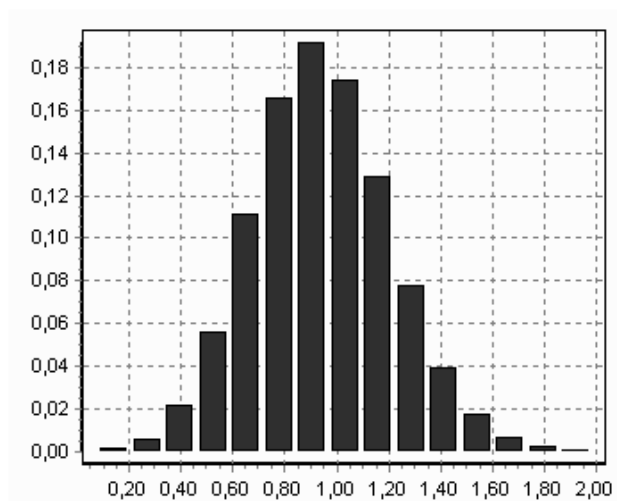


Рисунок 2.3 - Гистограмма
распределений максимального
уровня принятого сигнала ($b=12$ бит,
 $SNR=20$ дБ, $\sigma=0,2666$, $d(0,56;1,44)$)

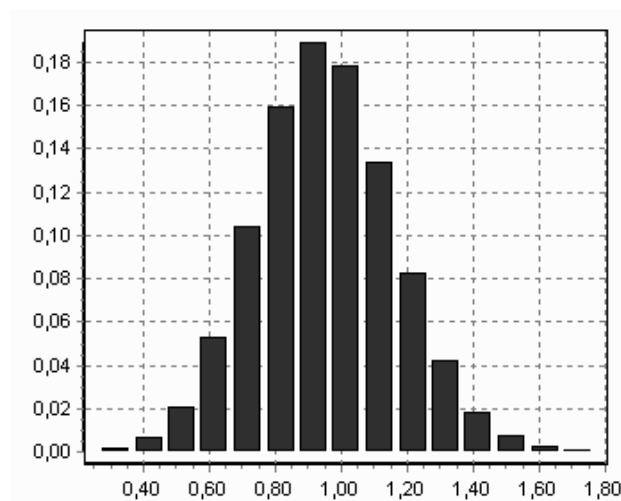


Рисунок 2.4 - Гистограмма
распределений максимального
уровня принятого сигнала ($b=12$ бит,
 $SNR=22$ дБ, $\sigma=0,2099$, $d(0,65;1,35)$)

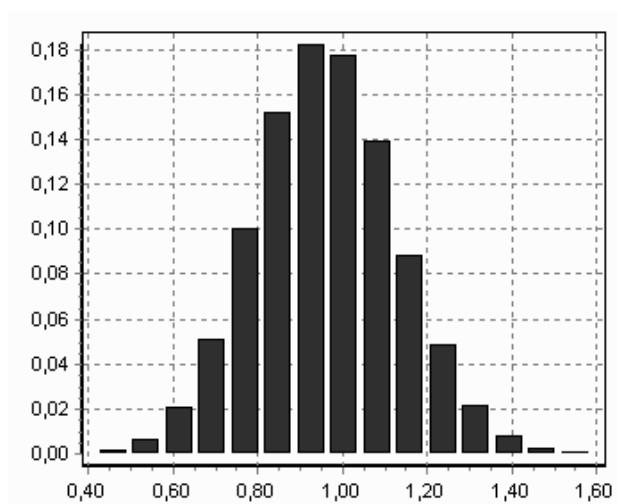


Рисунок 2.5 - Гистограмма
распределений максимального
уровня принятого сигнала ($b=12$ бит,
 $SNR=24$ дБ, $\sigma=0,1666$, $d(0,73;1,27)$)

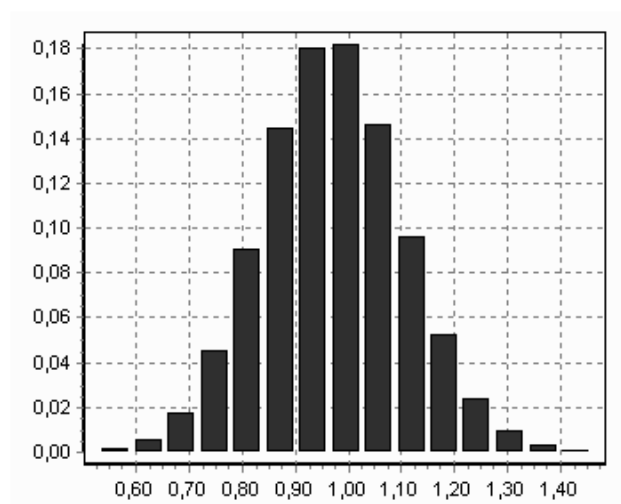


Рисунок 2.6 - Гистограмма
распределений максимального
уровня принятого сигнала ($b=12$ бит,
 $SNR=26$ дБ, $\sigma=0,1323$, $d(0,78;1,22)$)

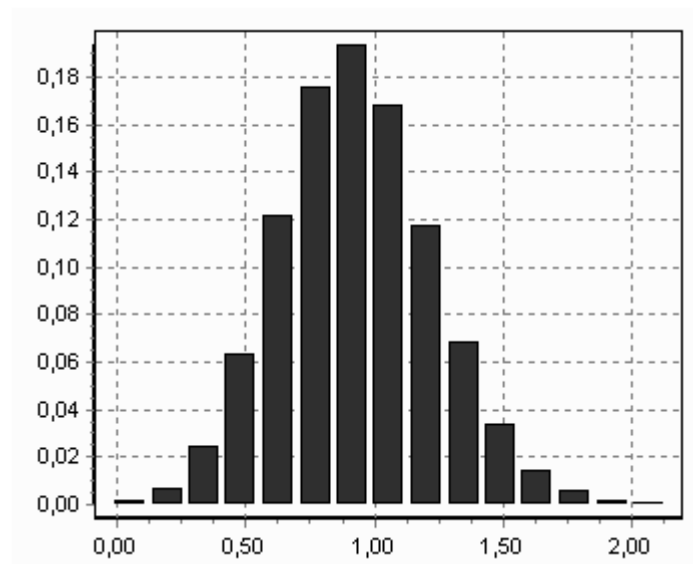


Рисунок 2.7 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=19$ дБ, $\sigma=0,2998$, $d(0,51;1,49)$). В стандартном алгоритме тоже значение σ ($\sigma=0,3003$) достигается при $SNR=20$ дБ)

П.3. Моделирование алгоритма коррекции КПХ, использующего в расчетах полярную систему координат с увеличенными по уровню значениями пилот-поднесущих

На рисунках 3.1-3.6 представлены гистограммы распределений максимального уровня принятого сигнала, где b – битность ЦАП, SNR – отношение сигнал/шум, σ – СКО, d – доверительный интервал.

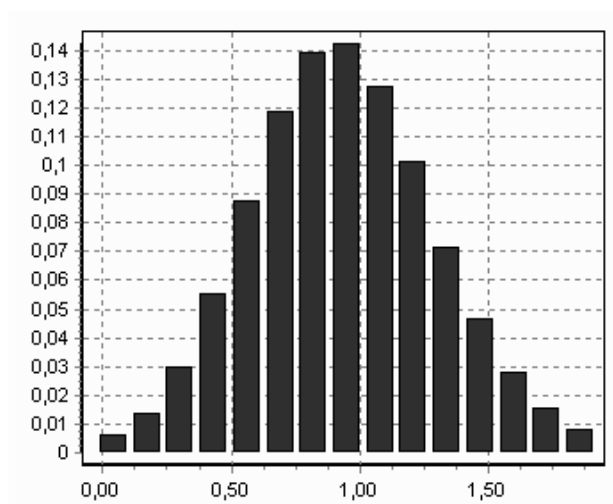


Рисунок 3.1 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=17$ дБ, $\sigma=0,3674$, $d(0,40;1,60)$)

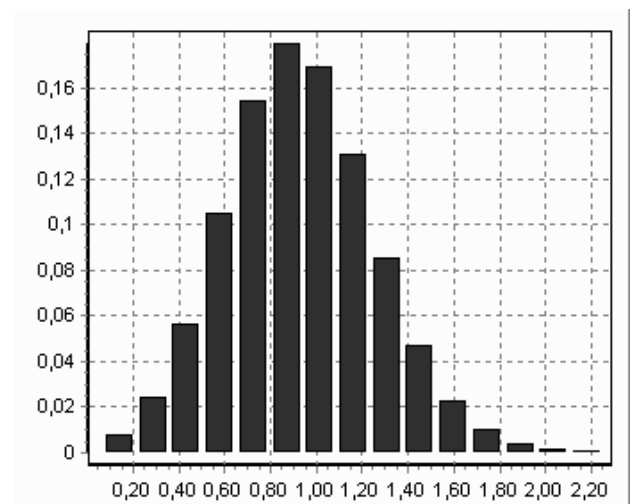


Рисунок 3.2 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=18$ дБ, $\sigma=0,3259$, $d(0,46;1,54)$)

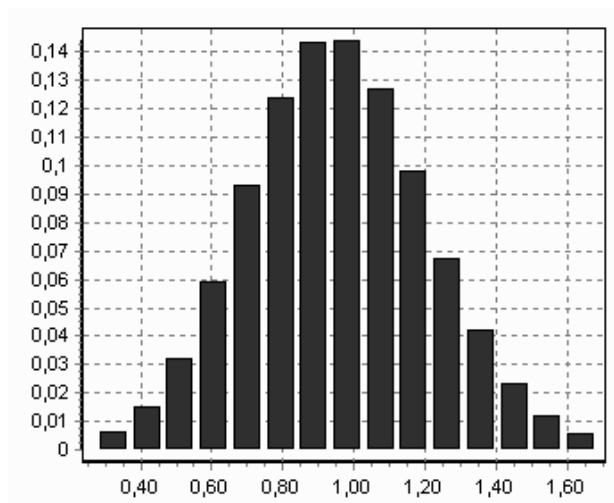


Рисунок 3.3 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=20$ дБ, $\sigma=0,2578$, $d(0,58;1,42)$)

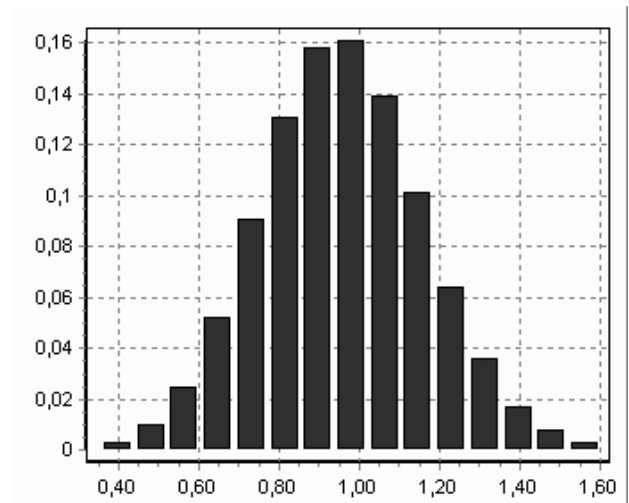


Рисунок 3.4 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=22$ дБ, $\sigma=0,2038$, $d(0,66;1,34)$)

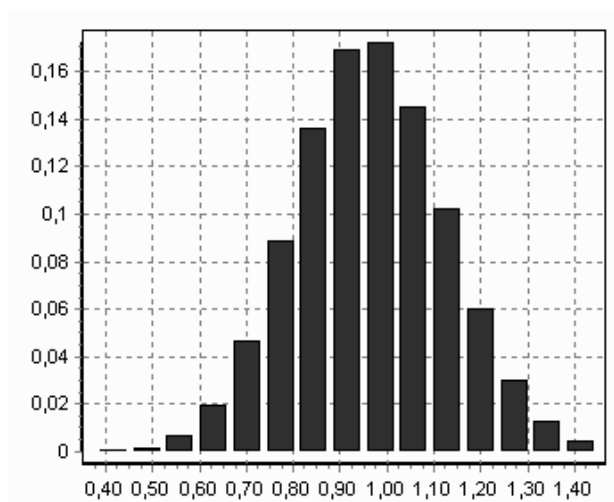


Рисунок 3.5 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=24$ дБ, $\sigma=0,1620$, $d(0,73;1,27)$)

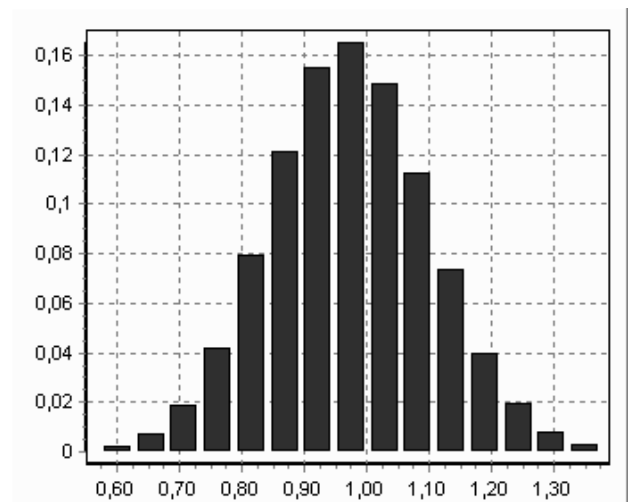


Рисунок 3.6 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=26$ дБ, $\sigma=0,1286$, $d(0,79;1,21)$)

С увеличением отношения сигнал/шум значение σ уменьшается. Найденные значения СКО будут использованы при сравнении алгоритмов.

П.4. Моделирование модифицированного алгоритма коррекции КПХ канала связи при различных значениях параметра «сдвиг от начала символа»

На рисунках 4.1-4.4 представлены гистограммы распределений максимального уровня принятого сигнала при различных значениях параметра «сдвиг от начала символа», который определяет момент от начала символа, когда начинается запись принятых данных в массивы. Для *OFDM* это значение не может превышать величину циклического префикса. Моделирование проведено для величины циклического префикса равной 32 дискрета и сдвига равного 1, 8, 24 и 32 дискрета. Заметим, что 32 дискрета соответствует приятию сигнала без циклического префикса. На рисунках обозначено: b – битность ЦАП, SNR – отношение сигнал/шум, σ – СКО, d – доверительный интервал.

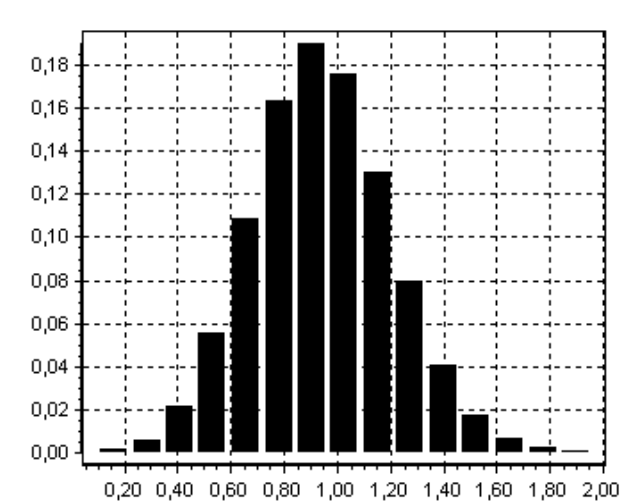


Рисунок 4.1 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=20$ дБ, $\sigma=0,2594$, $d(0,57;1,43)$, сдвиг от начала символа 1 дискрет)

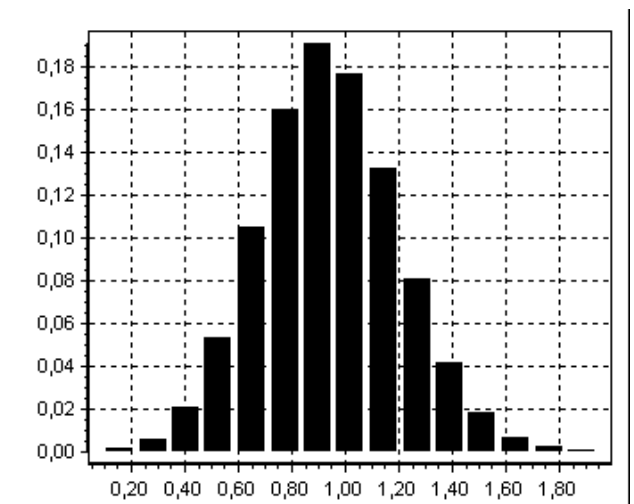


Рисунок 4.2 - Гистограмма распределений максимального уровня принятого сигнала ($b=12$ бит, $SNR=20$ дБ, $\sigma=0,2587$, $d(0,57;1,43)$, сдвиг от начала символа 8 дискрет)

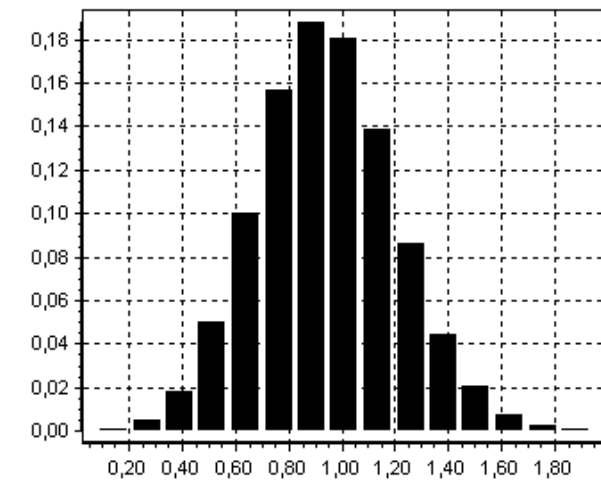


Рисунок 4.3 - Гистограмма
распределений максимального
уровня принятого сигнала ($b=12$ бит,
 $SNR=20$ дБ, $\sigma=0,2588$, d (0,57;1,43),
сдвиг от начала символа 24
дискрета)

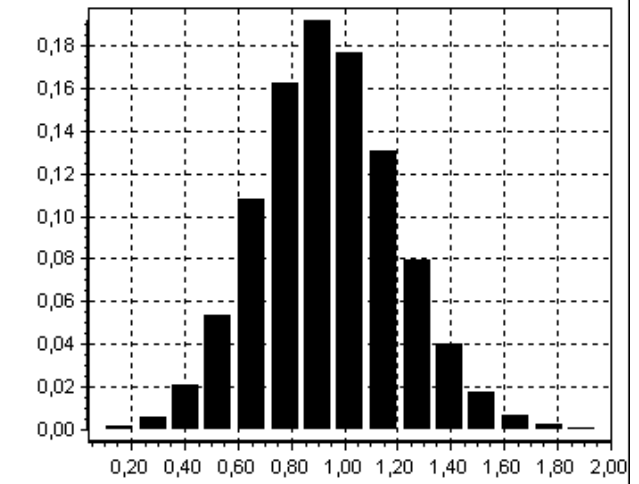


Рисунок 4.4 - Гистограмма
распределений максимального
уровня принятого сигнала ($b=12$ бит,
 $SNR=20$ дБ, $\sigma=0,2585$, d (0,57;1,43),
сдвиг от начала символа 32
дискрета)

Моделирование показало, что изменение задержки в пределах длины циклического префикса не влияет на качество коррекции спектра, поэтому неравномерность группового времени запаздывания в пределах длины циклического префикса также не будет влиять на работоспособность предложенных алгоритмов.

П.5. Заявка на изобретение 2015109441 «Способ временной синхронизации системы связи на основе ортогонального частотного разделения каналов с мультиплексированием по преамбуле»

РОССИЙСКАЯ ФЕДЕРАЦИЯ



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(19) RU (11)

2015109441 (13) A

(51) МПК

H04B7/26 (2006.01)

(12) ЗАЯВКА НА ИЗОБРЕТЕНИЕ

По данным на 16.11.2016 состояние делопроизводства: Экспертиза по существу завершена

(21), (22) Заявка: 2015109441, 17.03.2015

Приоритет(ы):

(22) Дата подачи заявки: 17.03.2015

(43) Дата публикации заявки: 10.10.2016

Адрес для переписки:

600015, г. Владимир, пр-кт Ленина, 39, кв. 38.

Позднякову В.А.

(71) Заявитель(и):

Поздняков Владислав Александрович (RU)

(72) Автор(ы):

Поздняков Владислав Александрович (RU),

Позднякова Лидия Васильевна (RU)

(54) Способ временной синхронизации системы связи на основе ортогонального частотного разделения каналов с мультиплексированием по преамбуле

(57) Формула изобретения

Способ временной синхронизации системы связи, заключающийся в том, что на передающей стороне формируют преамбулу из OFDM символов, модулированных по схеме квадратурной фазовой манипуляции, содержащих циклический префикс, первый символ которой используют для определения по нему временного положения сигнала; сигнал преамбулы переносят на несущую частоту; усиливают и передают его по каналу связи; на приемной стороне входной сигнал фильтруют, усиливают, переносят на нулевую промежуточную частоту, осуществляют его аналого-цифровое преобразование, формируя входной цифровой комплексный сигнал на нулевой промежуточной частоте; далее формируют решающую функцию и определяют ее максимум и по его положению оценивают временную задержку сигнала, отличающийся тем, что формируют символ преамбулы не имеющий повторяющихся участков; рассчитывают решающую функцию, как квадрат модуля комплексной функции взаимной корреляции между принятым сигналом и заранее известным символом эталонной преамбулы без циклического префикса; определяют участок значимости так, чтобы значение решающей функции на границах диапазона значимости было больше заданного уровня по отношению к максимуму решающей функции, участок значимости обязательно включает индекс положения максимума, а решающая функция за пределами участка значимости не имеет значений больше заданного уровня по отношению к максимуму решающей функции; рассчитывают сумму значений решающей функции на участке значимости; рассчитывают среднее значение решающей функции на всем ее протяжении за исключением участка значимости; используют основной критерий обнаружения преамбулы, заключающийся в том, что сумма значений решающей функции на участке значимости должна превышать произведение установленного порогового коэффициента на среднее значение решающей функции, рассчитанное при исключении участка значимости.

П.6. Свидетельство государственной регистрации программы для ЭВМ RU 2015618433

09.06.2016

Программа для ЭВМ №2015618433

РОССИЙСКАЯ ФЕДЕРАЦИЯ

RU 2015618433



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

ГОСУДАРСТВЕННАЯ РЕГИСТРАЦИЯ ПРОГРАММЫ ДЛЯ ЭВМ

Номер регистрации (свидетельства):
2015618433

Дата регистрации: 10.08.2015

Номер и дата поступления заявки:
2015615548 23.06.2015

Дата публикации: [20.09.2015](#)

Авторы:

Позднякова Лидия Васильевна (RU)

Правообладатель:

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых» (ВлГУ) (RU)

Название программы для ЭВМ:

Программа моделирования коррекции комплексной передаточной характеристики канала связи по преамбуле и пилот-поднесущим в соответствии со стандартом WiMAX 802.16.2004

Реферат:

Программа предназначена для моделирования алгоритма корректировки спектра принятого OFDM символа, реализованного на основе стандарта WiMAX 802.16.2004. Коррекция осуществляется с использованием в расчетах полярной системы координат и линейной интерполяции. Реализована возможность изменения значения частоты дискретизации, изменения размера циклического префикса, изменения уровня шумов, установки закона распределения шумов: Гаусса, Рэлея, Вейбулла, равномерного, фильтрации шумов в заданной полосе частот. Программа позволяет строить: гистограммы распределения шумов и распределений максимального уровня принятого сигнала, переданный сигнал с шумами, обратную комплексную передаточную характеристику, принятый сигнал, принятый сигнал после эквалайзера.

Тип реализующей ЭВМ: IBM PC-совмест. ПК

Язык программирования: Delphi

Вид и версия операционной системы: Windows 98 и выше

Объем программы для ЭВМ: 0,8 Мб

П.7. Свидетельство государственной регистрации программы для ЭВМ RU 2015618752

09.06.2016

Программа для ЭВМ №2015618752

РОССИЙСКАЯ ФЕДЕРАЦИЯ

RU 2015618752



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

ГОСУДАРСТВЕННАЯ РЕГИСТРАЦИЯ ПРОГРАММЫ ДЛЯ ЭВМ

Номер регистрации (свидетельства):
2015618752

Дата регистрации: 17.08.2015

Номер и дата поступления заявки:
2015615411 22.06.2015

Дата публикации: [20.09.2015](#)

Авторы:

Позднякова Лидия Васильевна (RU)

Правообладатель:

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых» (ВлГУ) (RU)

Название программы для ЭВМ:

Программа моделирования целочисленного алгоритма коррекции комплексной передаточной характеристики канала связи на основе ортогонального частотного разделения каналов с мультиплексированием (OFDM)

Реферат:

Программа предназначена для моделирования работы целочисленного алгоритма корректировки спектра принятого OFDM символа данных. Коррекция осуществляется в 2 этапа: по преамбуле и пилот-поднесущим OFDM сигнала. Алгоритм коррекции работает в ортогональной системе координат. Реализована возможность изменения размера циклического префикса, изменения значения частоты дискретизации, изменения уровня шумов, имеющих нормальный закон распределения, в заданной полосе частот. Программа позволяет строить: гистограммы распределения шума и распределений максимального уровня принятого сигнала, обратную комплексную передаточную характеристику, принятый сигнал после эквалайзера, принятый сигнал, переданный сигнал с шумом.

Тип реализующей ЭВМ: IBM PC - совмест. ПК

Язык программирования: Delphi

Вид и версия операционной системы: Windows 98 и выше

Объем программы для ЭВМ: 0,8 Мб

П.8. Свидетельство государственной регистрации программы для ЭВМ RU 2015618979

09.06.2016

Программа для ЭВМ №2015618979

РОССИЙСКАЯ ФЕДЕРАЦИЯ

RU 2015618979



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

ГОСУДАРСТВЕННАЯ РЕГИСТРАЦИЯ ПРОГРАММЫ ДЛЯ ЭВМ

Номер регистрации (свидетельства):
2015618979

Дата регистрации: 20.08.2015

Номер и дата поступления заявки:
2015615843 30.06.2015

Дата публикации: [20.09.2015](#)

Авторы:

Позднякова Лидия Васильевна (RU)

Правообладатель:

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых» (ВлГУ) (RU)

Название программы для ЭВМ:

Программа расчета коэффициентов преамбулы для системы связи на основе ортогонального частотного разделения каналов с мультиплексированием (OFDM) для поиска ее оптимального по пик-фактору варианта

Реферат:

Программа предназначена для расчета коэффициентов преамбулы для системы связи на основе ортогонального частотного разделения каналов с мультиплексированием (OFDM), с целью определения состава спектральных компонент преамбулы, имеющей наименьшее значение пик-фактора во временной области. Программа позволяет задавать количество компонент в преобразовании Фурье и количество используемых компонент. В программе отдельно рассчитываются спектральные компоненты для ВЧ и НЧ частей преамбулы. Программа предусматривает возможность их сохранения и объединения для получения единого текстового файла с набором спектральных компонент найденной преамбулы.

Тип реализующей ЭВМ: IBM PC - совмест. ПК

Язык программирования: Delphi

Вид и версия операционной системы: Windows 98 и выше

Объем программы для ЭВМ: 0,7 Мб

П.9. АКТ использования научных результатов диссертации в ООО «Предприятие по модернизации авиационных комплексов»



ООО "Предприятие по модернизации авиационных комплексов"
ИНН 7709533330 КПП 771401001
Банковские реквизиты: р/с 40702810938180007186 ПАО «Сбербанк России»
г. Москва, к/с 30101810400000000225 БИК 044525225
Юр. адрес: 123007, г. Москва, ул. 4-я Магистральная, д. 11
Тел./факс: + 7 (495) 518-87-37
E-mail: info@mak.aero

исх. № F2-160713-F02-060012-01

13.07.2016



УТВЕРЖДАЮ

Генеральный директор
Г.Г. Перуева

«13» июля 2016 г.

АКТ использования научных результатов диссертации

Поздняковой Лидии Васильевны на тему: "Развитие методов коррекции комплексной передаточной характеристики в системах с ортогональным частотным разделением и мультиплексированием", представленной на соискание учёной степени кандидата технических наук по специальности 05.12.04 - "Радиотехника, в том числе системы и устройства телевидения".

В ООО "Предприятие по модернизации авиационных комплексов" использованы следующие научные результаты диссертации Л.В. Поздняковой:

1. Алгоритм временной синхронизации канала связи OFDM при использовании преамбулы не имеющей повторяющихся участков во временной области, т.е. имеющей дискретно-непрерывный набор спектральных компонент.

2. Методика тактовой и частотной синхронизации за счет применения алгоритма временной синхронизации.

3. Целочисленный алгоритм компенсации влияния комплексной передаточной характеристики системы связи OFDM на основе преамбулы и пилот-поднесущих.

Разработанные соискателем алгоритмы позволили снизить требования к ресурсам ПЛИС используемой в OFDM модеме, а также повысили помехоустойчивость по сравнению со стандартными алгоритмами на 1.3 дБ.

Главный конструктор

Лавров М.С.

П.10. АКТ внедрения научных результатов диссертации в ФГБОУ «ВлГУ»

УТВЕРЖДАЮ:
Первый проректор, проректор по НиИР
доктор технических наук, профессор
В.Г. Прокошев
«15» сентября 2016 г.



Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

ул. Горького, д.87, г. Владимир, Россия, 600000, Телефон 4922532575, e-mail: oid@vlsu.ru

АКТ

внедрения научных результатов диссертации

Поздняковой Лидии Васильевны на тему: «Развитие методов коррекции комплексной передаточной характеристики в системах с ортогональным частотным разделением каналов и мультиплексированием», представленной на соискание учёной степени кандидата наук по специальности 05.12.04 – Радиотехника, в том числе системы и устройства телевидения.

В учебном процессе ВлГУ на кафедре Радиотехники и Радиосистем использованы следующие научные результаты диссертации Л.В. Поздняковой:

1. Алгоритм коррекции комплексной передаточной характеристики канала связи с ортогональным частотным разделением каналов и мультиплексированием (*OFDM*).
2. Модель канала связи *OFDM* с аддитивным белым Гауссовым шумом.

Для ОС Windows создана программа моделирования, корректирующая спектр принятого *OFDM* символа по преамбуле и пилот-поднесущим, на основе которой разработана лабораторная работа по курсу «Радиоприемные устройства» для направления 210400 «Радиотехника».

Заведующий кафедрой РТ и РС,
Доктор технических наук, профессор
заслуженный деятель науки РФ

 О.Р. Никитин