

Санкт-Петербургский государственный электротехнический университет  
«ЛЭТИ» им. В.И.Ульянова (Ленина)  
(СПбГЭТУ «ЛЭТИ»)

Специальность	11.04.02 – Инфокоммуникационные техно- логии и системы связи
Специализация	Интеллектуальные инфокоммуникацион- ные сети
Факультет	РТ
Кафедра	РС
<i>К защите допустить</i>	
Зав. кафедрой	

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
МАГИСТРА**

**Тема: ПРОГРАММНЫЙ МОДУЛЬ ОБРАБОТКИ СИГНАЛОВ  
СТАНДАРТА DMR**

Студент	_____	Боржонов А.И.
	<i>Подпись</i>	
Руководитель	_____	Андреева О.М.
	<i>(Уч. степень, уч. звание) Подпись</i>	
Консультанты	_____	<b>Иванов И.И.</b>
	<i>(Уч. степень, уч. звание) Подпись</i>	
	_____	<b>Иванов И.И.</b>
	<i>(Уч. степень, уч. звание) Подпись</i>	
	К.Т.Н., доц.	Маругин А. С.
	<i>(Уч. степень, уч. звание) Подпись</i>	

Санкт-Петербург  
2024

## ЗАДАНИЕ

### НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ

Утверждаю  
зав. кафедрой РС

«        »        2024 г.

Студент      Боржонов А.И.      Группа 8182

# Тема работы: Программный модуль обработки сигналов стандарта DMR

Место выполнения ВКР: ООО «СТЦ»

Исходные данные (технические требования): Стандарт DMR, язык программирования C++, реализация с использованием объектно-ориентированного стиля программирования, АЧХ фильтров демодулятора, использование библиотеки цифровой обработки сигнала ООО «СТЦ», быстродействие алгоритма не хуже чем 1:1, вероятность ложной тревоги при обнаружении  $10^{-4}$

## Содержание ВКР:

Введение, Цифровая связь, Радиомониторинг, Стандарт DMR, Алгоритм обработки сигнала, Специальные вопросы обеспечения безопасности, Заключение.

Перечень отчетных материалов: текст ВКР, иллюстративный материал.

## Дополнительные разделы: Специальные вопросы обеспечения безопасности

Дата выдачи задания  
«        » 2024 г.

Дата представления ВКР к защите  
«    »    2024 г.

Студент Боржонов А.И.

Руководитель \_\_\_\_\_ Андреева О.М.  
(Уч. степень, уч. звание)

Консультант Литвиненко С.А.

# КАЛЕНДАРНЫЙ ПЛАН ВЫПОЛНЕНИЯ ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ

Утверждаю  
Зав. кафедрой РС

«\_\_» \_\_\_\_\_ 2024 г.

Студент Боржонов А.И.

Группа 8182

Тема работы: Программный модуль обработки сигналов стандарта DMR

п/п	Наименование работ	Срок выполнения
1	Обзор литературы по теме работы	00.00 – 00.00
2	Реализация отдельных этапов алгоритма в Matlab	00.00 – 00.00
3	Реализация алгоритма на языке программирования C++	00.00 – 00.00
4	Отладка программного обеспечения на устройстве	00.00 – 00.00
5	Оформление пояснительной записки	00.00 – 00.00
6	Оформление иллюстративного материала	

Студент

\_\_\_\_\_

Боржонов А.И.

Руководитель

\_\_\_\_\_

Андреева О.М.

Литвиненко С.А.

\_\_\_\_\_  
(Уч. степень, уч. звание)

## РЕФЕРАТ

Пояснительная записка 00 стр., 00 рис., 00 табл., 00 ист., 00 прил.

### ЦИФРОВАЯ СВЯЗЬ, РАДИОМОНИТОРИНГ, СТАНДАРТА DMR

Цель работы – реализация программного модуля обработки сигналов стандарта DMR.

В работе представлены теоретические сведения, касающиеся вопросов цифровой связи и радиомониторинга, рассмотрены основные положения стандарта TETRA, необходимые для реализации алгоритма обработки сигнала стандарта DMR.

Реализация программного модуля представляет собой кроссплатформенное серверное программное обеспечение, реализованное на языке программирования C++ с использованием объектно-ориентированного стиля программирования.

К результатам данной работы относится программное обеспечение высокой производительности, реализующее сканирование заданного частотного диапазона на предмет наличия устройств стандарта DMR и их распознавания.

## **ABSTRACT**

The object of development is a software module for signal processing of the DMR standard.

The purpose of the work is to implement a software module for signal processing of the DMR standard.

The paper presents theoretical information on issues of digital communication and radio monitoring, as well as the main provisions of the TETRA standard necessary for the implementation of the DMR signal processing algorithm.

The implementation of the software module is a cross-platform server software implemented in the C++ programming language using an object-oriented programming style.

The results of this work include high-performance software that implements scanning of a given frequency range for the presence of DMR standard devices.

## СОДЕРЖАНИЕ

	Введение	
1	Цифровая связь	
1.1	Общая структурная схема и основные характеристики систем цифровой связи	
1.2	Множественный доступ	
1.3	Помехоустойчивое кодирование	
2	Радиомониторинг	
2.1	Задачи средств радиомониторинга	
2.2	Классификация и структура средств радиомониторинга	
2.3	Пеленгование источников радиоизлучения	
3	Стандарт DMR	
3.1	Структура TDMA фрейма	
3.2	Каналы управления DMR	
3.3	Физический уровень DMR	
4	Алгоритм обработки сигнала	
4.1	Устройство АРМ	
4.2	Основные этапы алгоритма	
4.3	Демодуляция	
4.4	Обнаружение	
4.5	Декодирование	
4.6	Быстродействие алгоритма	
4.7	Помехоустойчивость алгоритма	
5	Специальные вопросы обеспечения безопасности	
5.1	Используемое в разработке программное обеспечение	
5.2	Удобство и пригодность ПО	
5.3	Выводы по 5 главе	

Заключение

Список использованных источников

69

## ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

В настоящей пояснительной записке применяют следующие термины с соответствующими определениями:

АРМ – автоматизированный радиомониторинг

БС – базовая станция

ВКФ – взаимная корреляционная функция

ИРИ – источник радиоизлучения

ОСШ – отношение сигнал/шум

ПО – программное обеспечение

ПУ – приемное устройство

УЦОД – устройство цифровой обработки данных

BPTC – Block Product Turbo Code

CRC – Cyclic Redundancy Check

CSBK – Control Signalling Block

DMR – Digital Mobile Radio

ETSI – European Telecommunications Standards Institute

FEC – Forward Error Correction

FID – Feature set ID

FM – Frequency Modulation

ID – Identifier

LC – Link Control

MBC – Multi Block Control

MS – Mobile Station

RC – Reverse Channel

SYNC – SYNChronization

TDMA – time division multiple access

TX – Transmitted bit

4FSK – Four-level Frequency Shift Keying

## ВВЕДЕНИЕ

Digital Mobile Radio (DMR) – это открытый стандарт цифровой радиосвязи, разработанный Европейским институтом телекоммуникационных стандартов (European Telecommunications Standards Institute (ETSI)). Открытый стандарт цифровой мобильной радиосвязи DMR появился в 2005 году. Его требования состояли из трех частей, которые появлялись постепенно в течение последующих лет, так что производство радиостанций, полностью соответствующих стандарту, началось в 2007 году. Радиостанции DMR работают в частотных диапазонах ОВЧ/СВЧ, то есть 136-174/403-470 МГц.

Популярность и открытость стандарта привели к тому, что количество новых базовых станций (БС) постоянно увеличивается. Эти БС должны соответствовать ряду правил и ограничений, регламентированных законодательством. Задачу контроля за соблюдением этих правил помогают выполнять средства автоматизированного радиомониторинга.

Целью данной работы является реализация программного модуля обработки сигнала стандарта DMR. Этот модуль является неотъемлемой частью программного обеспечения (ПО) устройства автоматизированного радиомониторинга (АРМ) (рис. 1), выполняющего сканирование частотного диапазона для обнаружения, идентификации и пеленгования источников радиоизлучения (ИРИ).

При реализации программного модуля был задан ряд ограничений:

- использование класса *Correlator* из библиотеки цифровой обработки сигнала ООО «СТЦ»;
- использование ранее рассчитанных фильтров для демодуляции сигнала DMR;
- требования по быстродействию должны быть выполнены на заданном устройстве АРМ.

Для достижения поставленной цели необходимо решить следующие задачи:



1. обзор литературы по вопросам цифровой связи и радиомониторинга (РМ);
2. изучение стандарта DMR Air Interface protocol;
3. изучение библиотеки цифровой обработки сигнала ООО «СТЦ»;
4. изучение основных этапов формирования сигнала физического уровня;
5. разработка алгоритма обработки сигнала;
6. разработка алгоритмов демодуляции;
7. разработка алгоритмов декодирования;
8. реализация алгоритма на языке программирования C++;
9. верификация алгоритма;
10. тестирование ПО на устройстве.

Основными требованиями к программному модулю являются использование объектно-ориентированного стиля программирования, высокая скорость работы алгоритма приема (время работы менее 10 мс) и вероятность ложной тревоги при обнаружении сигналов (менее  $10^{-4}$ ).

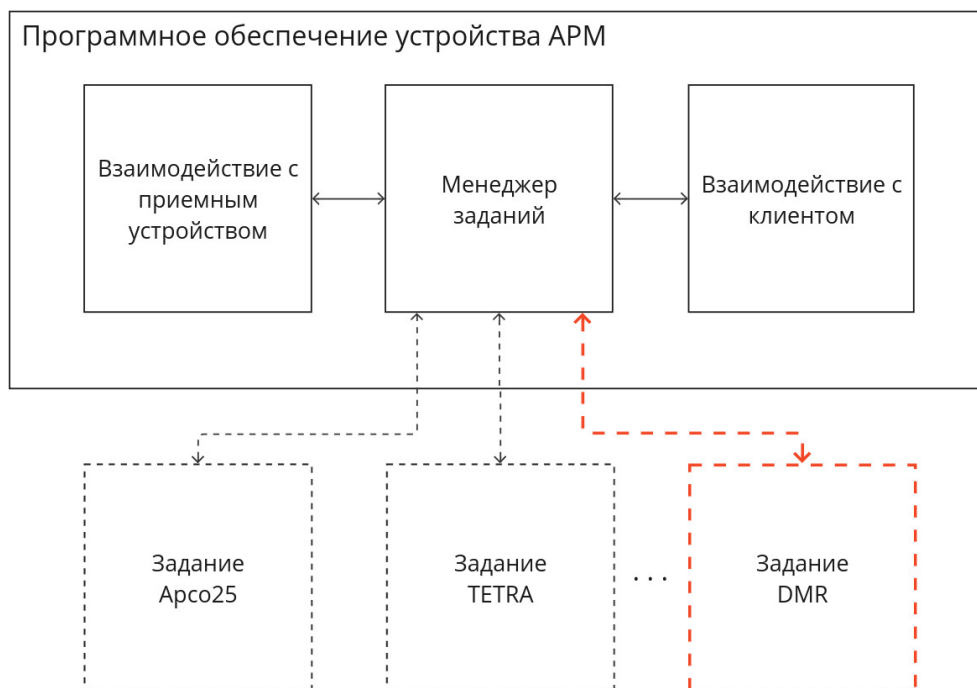


Рисунок 1 – Структурная схема ПО АРМ.

# 1. ЦИФРОВАЯ СВЯЗЬ

## 1.1. Общая структурная схема и основные характеристики систем цифровой связи.

Цифровая связь (ЦС) – это передача сообщений, представляющих собой элементы конечного множества [1]. Отличительной особенностью систем цифровой связи является то, что за конечный промежуток времени они посылают сигнал, состоящий из конечного набора элементарных сигналов. Для эффективной передачи сообщений, над ними производятся многократные преобразования. Общая структура системы цифровой связи показана на рис. 1.1.

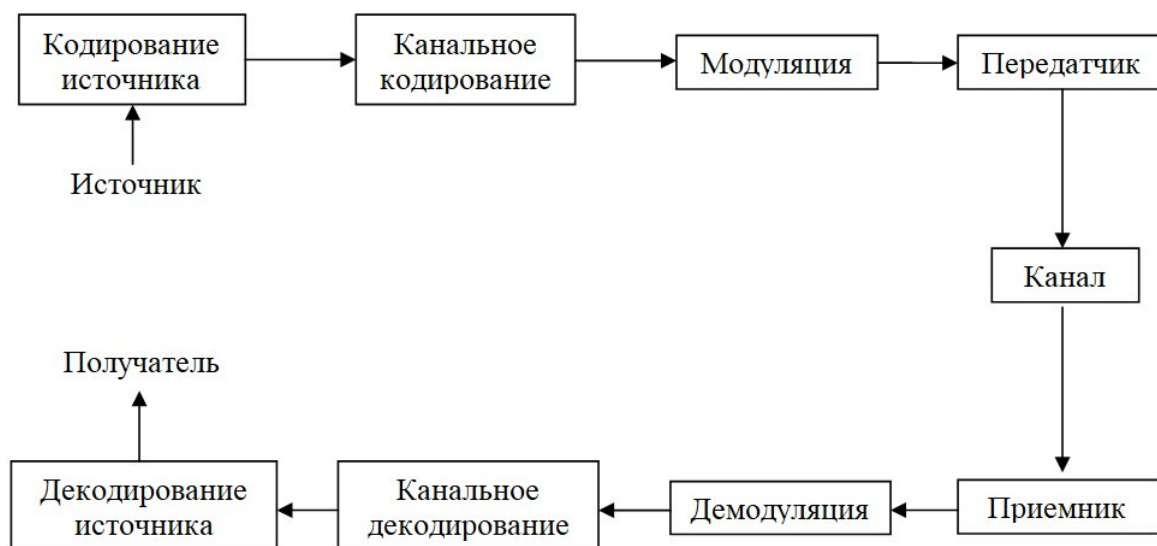


Рисунок 1.1 – Структурная схема цифровой связи.

Преобразование выхода источника, будь то аналогового или цифрового, в последовательность двоичных символов называется кодированием источника.

Следующий этап – канальное кодирование. Цель кодера канала состоит в том, чтобы ввести управляемым способом избыточность в цифровой поток, которая может использоваться в приёмнике для учета влияния шума и интерференции, возникающей при передаче сообщения через канал [3].

Далее сообщение поступает в модулятор, преобразующий цифровое сообщение в аналоговый модулированный сигнал.

В передатчике осуществляется усиление, перенос спектра модулированного сигнала в заданную полосу частот и передача сигнала по каналу связи.

Канал связи представляет собой физическую среду, которая используется для передачи сигнала от передатчика к приёмнику.

Приемник осуществляет предварительную обработку полученного аналогового сигнала. В состав приемника обычно входит гетеродин, предназначенный для переноса сигнала с несущей на промежуточную частоту.

Затем демодулятор обрабатывает искажённый каналом сигнал и преобразует его в последовательность чисел, представляющих собой оценки переданных данных.

Далее производится канальное декодирование или декодирование помехоустойчивого кода, при этом, благодаря корректирующим свойствам кода, возможно исправление части ошибок, возникших в процессе передачи [1].

На последнем этапе декодер источника получает выходную последовательность от декодера канала и восстанавливает исходную форму сигнала.

Одним из наиболее важных понятий в цифровой связи является понятие битов и символов.

Бит – это единица измерения количества информации, а также элемент входного потока данных, который может принимать значения 0 и 1.

Символ – группа битов, предназначенная для передачи большего количества информации. Количество битов в символе обозначается буквой  $m$ , а количество значений, которые может принимать символ, – буквой  $M$ , так что  $M = 2^m$ .

Цифровые системы связи характеризуются рядом технических параметров, наиболее важными из которых являются:

- Скорость передачи данных:

- битовая – число бит, переданных в единицу времени —  $R$ , бит/с;
- символьная – число символов, переданных в единицу времени –  $R_{sym} = R/m$ , симв./с, где  $m$  – число бит в символе.
- Ширина спектра сигнала –  $W$ , Гц. Подразумевается измерение ширины по какому-либо практическому критерию.
- Спектральная эффективность –  $R/W$ , (бит/с)/Гц, показывающая насколько эффективно система использует полосу частот для передачи информации.
- Вероятность ошибки, т. е. вероятность того, что принятое числовое значение не совпадает с переданным:
  - битовая – вероятность того, что принятое значение бита, не совпадает с переданным, обозначается как  $P_b$  или Bit Error Rate;
  - символьная – вероятность того, что принятое значение символа не совпадает с переданным, обозначается как  $P_{sym}$  или Symbol Error Rate (SER),  $P_{sym} \in [0, (M-1)/M]$ .
- Отношение сигнал/шум (ОСШ) в системах цифровой связи может измеряться по-разному:
  - ОСШ по мощности (signal noise ratio (SNR)), дБ. Показывает отношение мощности сигнала к мощности шума.
  - ОСШ на бит, определяемое как отношение энергии сигнала  $E_b$ , приходящейся на один бит передаваемых данных, к спектральной плотности мощности белого шума –  $E_b/N_0$ , дБ.

## 1.2. Цифровая модуляция

Цифровая модуляция – это процесс преобразования цифровых символов в сигналы, совместимые с характеристиками канала [2]. Для цифровых

систем различают две разновидности модуляции – низкочастотную (baseband) и полосовую (passband).

В первом случае формируется низкочастотный сигнал имеющий вид импульсов заданной формы. Во втором случае цифровое сообщение используется для управления параметрами несущего колебания (как правило, синусоидального), таким образом, получается сигнал со спектром, сосредоточенным в окрестностях заданной несущей частоты [1].

Полосовая модуляция является необходимым этапом для всех систем, использующих радиопередачу. Это связано с тем, что при передаче сигнала в среде используются антенны, размер которых зависит от длины волны передаваемых колебаний. Так, при передаче низкочастотных сигналов размер антенны может достигать нескольких километров, а при передаче высокочастотных сигналов - не более нескольких сантиметров.

Еще одним важным преимуществом полосовой модуляции является разделение различных сигналов, это помогает минимизировать последствия интерференции.

Основные типы полосовой модуляции перечислены на рис. 1.2.



Рисунок 1.2 – Основные типы полосовой модуляции.

Если для обнаружения сигнала приемнику требуется информация о фазе несущего колебания, процесс называется когерентным обнаружением, если такая информация не используется, он называется некогерентным обнаружением.

ружением, преимуществом некогерентных систем перед когерентными является простота, а недостатком – большая вероятность ошибки.

### **1.3. Множественный доступ**

Ресурс связи представляет время и ширину полосы частот, доступные для передачи сигнала в определенной системе [5]. Эффективное распределение ресурса связи является одним из основных способов повышения пропускной способности. Ниже перечислены основные методы распределения ресурса связи:

- Частотное разделение – разделение доступной полосы частот на фиксированное число непересекающихся подполос, которые должны быть назначены каждому пользователю системы связи.
- Разделение по времени – назначение периодических временных интервалов для пользователей. В различных системах время доступа пользователя к ресурсу может определяться динамически или быть заранее сконфигурированным.
- Кодовое разделение – выделение определенных элементов набора ортогонально (либо почти ортогонально) распределенных спектральных кодов, каждый из которых использует весь диапазон.
- Пространственное разделение – использование антенн с узкими диаграммами направленности для разделения и направления радиосигналов в разных направлениях. Этот метод позволяет многократно использовать одну полосу частот.
- Поляризационное разделение – использование ортогональной поляризации для разделения сигналов, что позволяет использовать один частотный диапазон.

#### **1.4. Помехоустойчивое кодирование**

При передаче цифровой информации по каналу связи из-за наличия шумов и помех существует некоторая вероятность искажения передаваемых символов [1]. Помехоустойчивое кодирование, также известное как канальное кодирование, используется для обнаружения или исправления возникающих ошибок.

Принцип помехоустойчивого кодирования заключается в добавлении контролируемой избыточности к сообщению. Добавленные избыточные символы связаны с символами сообщения через определенные математические соотношения. Благодаря избыточности кодированные сообщения максимально отличаются друг от друга. В результате добавления избыточности принимаются не все возможные последовательности символов, а только те, которые соответствуют используемому алгоритму кодирования. Таким образом, ошибки могут быть обнаружены и исправлены при получении искаженной информации.

## **2. РАДИОМОНИТОРИНГ**

### **2.1. Задачи средств радиомониторинга**

Радиомониторинг – это деятельность по изучению и контролю радиобстановки [4]. Радиомониторинг поддерживает процесс общего управления использованием спектра, предоставляя информацию о его фактическом использовании на основании данных измерений, степени загрузки каналов и полос радиочастот, а также данных о статистике доступности каналов и эффективности методов управления использованием спектра [5].

Основная цель радиомониторинга – получение неизвестной информации об ИРИ. Задачи, решаемые с помощью средств АРМ, включают в себя:

- непрерывный или периодический мониторинг широкого диапазона частот;
- обнаружение и анализ радиоизлучения для выявления источников сигналов и помех;
- измерение параметров сигналов и помех;
- оценка опасности или ценности для пользователей;
- определение местоположения источников наземных радиосигналов и источников помех;
- проведение работ по противодействию вредным ИРИ.

### **2.2. Классификация и структура средств радиомониторинга**

Средства АРМ представляют собой информационно-поисковые системы. Каждая из задач, описанных в разделе 2.1, является многоэтапной и требует использования различных радиотехнических средств для выполнения конкретных функций. Эти средства выполняют широкий спектр функций, включая анализ спектра в реальном времени, поиск широкополосного радиоизлучения, ведение базы данных ИРИ, контроль ИРИ, запись и воспроизведение радиосигналов, технический анализ ИРИ, пеленгацию ИРИ для различ-



ных видов модуляции, поиск и обнаружение технических каналов утечки информации и др.

Устройства АРМ можно разделить на категории в зависимости от зоны покрытия, области применения, выполняемых функций и производительности устройства.

Производительность устройств АРМ зависит от скорости панорамного спектрального анализа сигнала при заданных разрешении и динамическим диапазоне. Классификация средств АРМ по производительности приведена в таблице 2.1.

Таблица 2.1 – Классификация производительности средств АРМ

Производительность	Скорость анализа
Низкая	10...100 МГц/с
Средняя	100...1000 МГц/с
Высокая	1...10 МГц/с
Сверхвысокая	Выше 10 МГц/с

В зависимости от величины зоны действия средств АРМ их можно разделить на группы, например, используемые для решения задач радиомониторинга на местности, на внешних границах контролируемых объектов, в помещениях и т.д.

В зависимости от характера использования средства АРМ можно разделить на три основные группы: средства для открытого использования на стационарных и временных постах, средства, обеспечивающие скрытное использование, и средства, которые могут использоваться совместно (открыто или скрытно).

Современные устройства АРМ представляет собой систему аппаратно-программных средств, используемых для решения конкретной задачи. Ограничения на проектирование устройств АРМ включают в себя условия его использования и основные тактико-технические характеристики. При разра-

ботке средств АРМ необходимо соблюсти требования по функциональным возможностям, максимальным массогабаритным характеристикам и максимальной стоимости оборудования.

Таким образом, все средства радиомониторинга делятся на следующие группы [4, 5]:

- стационарные средства РМ;
- мобильные средства РМ наземного, воздушного и морского исполнения;
- портативные средства РМ;
- носимые средства РМ, предполагающее скрытое или открытое использование, так же предназначенные для выполнения задач во время движения оператора;
- измерительные средства для обеспечения контроля эффективности принятых мер по защите утечки информации, а также для измерения параметров излучений штатных радиосредств.

### **2.3. Пеленгование источников радиоизлучения**

Пеленг – это угол между направлением на ИРИ (вершина которого является точкой пеленгования) и некоторым опорным направлением. Если пеленг измеряется от северного географического меридиана, он называется истинным азимутом. Если в качестве исходного направления используется продольная ось транспортного средства, то пеленг называется относительным пеленгом (бортовым) (рис. 2.1)[4].

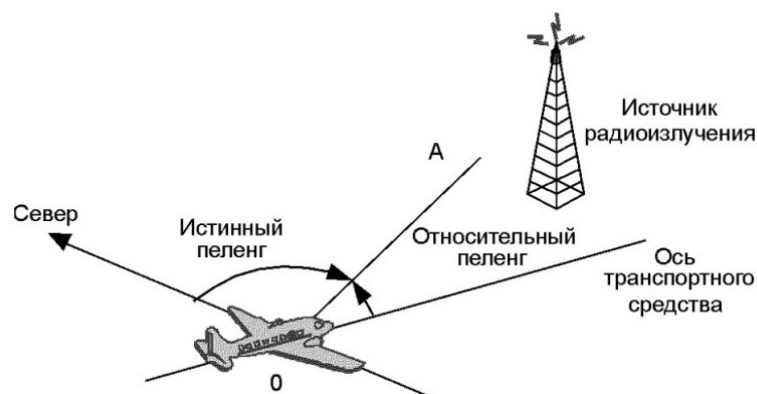


Рисунок 2.1 – Истинный и относительный пеленги.

Приборы для измерения пеленга – это устройства, измеряющие угол прихода радиосигналов и позволяющие определить направление на ИРИ. Основная задача навигационной системы – определение положения движущегося объекта (например, корабля или самолета) в земной системе координат.

В современных пеленгаторах используется технология цифровой обработки сигнала, благодаря чему отсутствует температурный дрейф и синхронизируется обработка сигналов разных каналов. Технические характеристики цифровых пеленгаторов более привлекательны по сравнению с аналоговыми.

Наиболее важные показатели качества пеленгаторов:

- точность пеленгования;
- чувствительность;
- помехоустойчивость;
- быстродействие;
- разрешающая способность;
- диапазон рабочих частот;
- вид пеленгуемого сигнала;
- время развёртывания;
- масса и габаритные размеры;
- сложность в производстве и эксплуатации;

- стоимость.

Один пеленгатор может определить только пеленг ИРИ. Для определения положения ИРИ требуется не менее двух дальних пеленгаторов. Положение ИРИ определяется пересечением двух пеленгов. Пеленги могут быть получены одновременно или последовательно.

Поскольку поверхность Земли не является плоской, пеленги могут быть представлены только прямыми линиями на относительно небольших расстояниях. В средних широтах эти расстояния примерно соответствуют линиям визирования. На больших расстояниях они отображаются на картах как геодезические линии (ортогональные линии), соединяющие любую точку на поверхности Земли кратчайшим путем. Вид геодезических линий зависит от используемого метода картографической проекции.

### 3. СТАНДАРТ DMR

В данной главе представлено подробное описание основных положений стандарта DMR, необходимых для реализации программного модуля.

#### 3.1. Структура TDMA фрейма

Система цифровой подвижной радиосвязи DMR применяет технологию многостанционного доступа с временным разделением каналов Time Division Multiple Access (TDMA) и полосу радиочастотной несущей, равной 12,5 кГц. Пакет DMR – период радиочастотной несущей, которая модулируется потоком данных. Поэтому пакет представлен тайм-слотом физического канала. Физический канал подсистемы DMR требуется, чтобы поддерживать логические каналы [6].

Логические каналы могут быть разделены на две категории:

- каналы информационного обмена, передающие речь или данные;
- каналы управления, передающие служебную информацию.

Обобщенная временная диаграмма обмена между абонентской (MS) и базовой (BS) станцией представлена на рис. 3.1, где слоты двух физических каналов TDMA маркированы «1» и «2». Восходящая посылка маркирована «MS TX», а нисходящая посылка маркирована «BS TX».

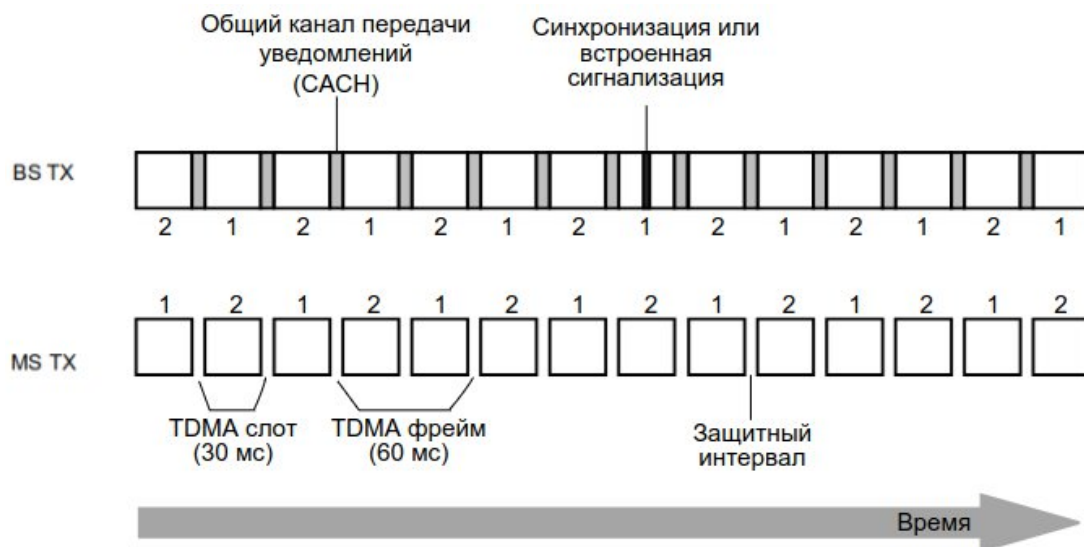


Рисунок 3.1 – Временная диаграмма обмена данными в DMR.

Пока БС остается активной, передача данных в нисходящем канале происходит непрерывно, вне зависимости от наличия данных для передачи. Передача данных в восходящем канале прекращается, когда на мобильной станции отсутствует информация для передачи. Для компенсации задержки времени распространения сигнала и времени нарастания сигнала усилителя в восходящем канале имеется неиспользуемый защитный временной интервал между пакетами.

Для управления каналом трафика в нисходящем, канале между пакетами содержится общий канал передачи уведомлений (Common Announcement Channel), также он содержит в себе низкоскоростную сигнализацию.

Размещение встроенной сигнализации в центре пакета позволяет во время передачи от абонентской станции дополнительно переходить к нисходящему каналу и восстанавливать информацию о реверсивном канале. Центры восходящих и нисходящих пакетов должны быть выравнены по времени [6].

Канальные пакеты 1 и 2 в нисходящем канале смещены на 30 мс относительно канальных пакетов 1 и 2 в восходящем канале. Такая схема передачи позволяет использовать одно поле идентификатора общего канала передачи уведомлений в нисходящем канале при обращении к восходящему и нисходящему каналу с тем же номером.

В речевых пакетах и пакетах данных используются различные сикхро-группы, чтобы позволить получателю различать пакеты. Различные сикхро-группы используются для восходящих и нисходящих каналов, чтобы помочь получателю исключать помехи от соседних каналов [7].

Расположение пакетов синхронизации в канале 1 не зависит от расположения пакетов синхронизации в канале 2. Расположение пакетов синхронизации в восходящих каналах не зависит от расположения пакетов в восходящих каналах [7].

Универсальная структура пакета состоит из двух 108-битных информационных полей и 48-битной синхронизации или поля сигнализации как пока-

зано на рис. 3.2. У каждого пакета есть общая длина 30 мс, но лишь 27,5 мс имеют 264 бита, которых достаточно для передачи 60 мс сжатой речи, используя 216 бит информационного поля.

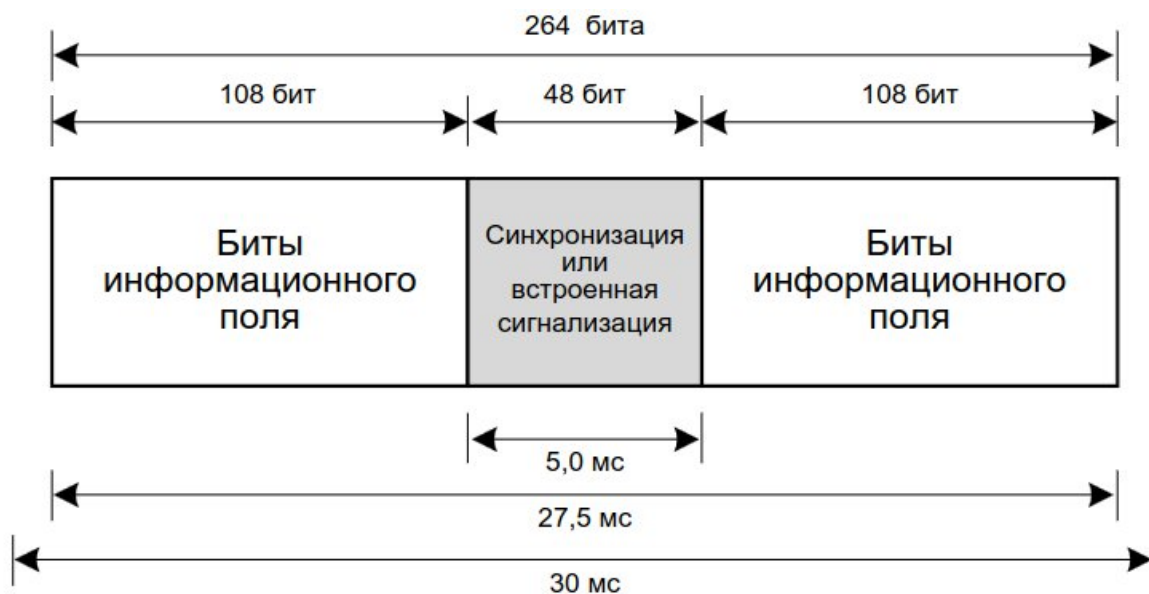


Рисунок 3.2 – Универсальная структура пакета.

В центре каждого пакета имеется поле, которое содержит синхрогруппу или встроенную сигнализацию. Это поле помещено в середине пакета, что необходимо для реализации реверсивного канала сигнализации.

В восходящем канале оставшиеся 2,5 мс используются в качестве защитного интервала для компенсации задержки распространения сигнала и времени нарастания сигнала усилителя (рис. 3.3).

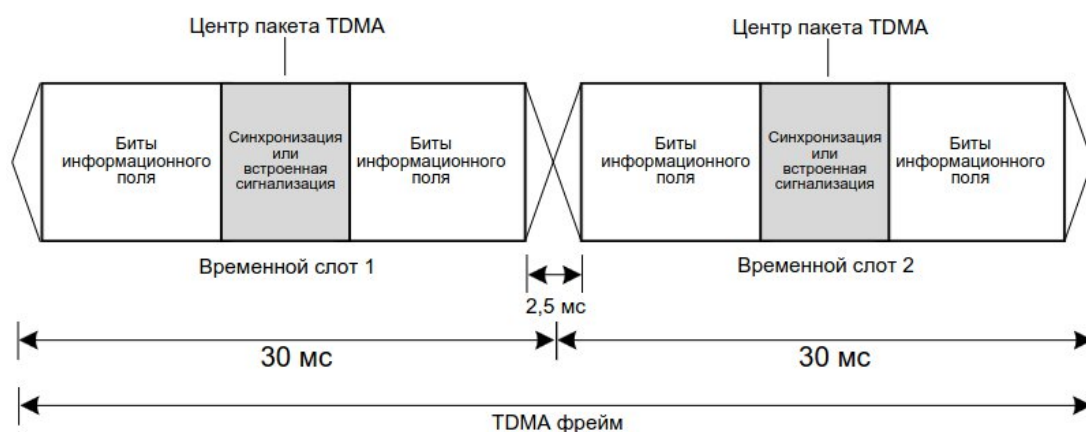


Рисунок 3.3 – Фрейм TDMA абонентской станции.

### **3.2. Каналы управления DMR**

Для сигнализации в стандарте DMR используются 4 типа сообщений:

- Link Control (LC) message – сообщение управления соединением;
- Control Signalling Block (CSBK) message – сообщение блока управления;
- Idle message – пустое сообщение;
- Multi Block Control (MBC) message – сообщение многоблочного управления.

#### **Структура LC сообщений**

Для сигнализации LC определено полное сообщение управления соединением (Full LC) и короткое сообщение управления соединением (Short LC). Короткое сообщение управлением соединением не содержит в себе специфической информации, позволяющей однозначно идентифицировать отправителя и получателя сообщения, поэтому его структура не будет рассмотрена в дальнейшем.

Сообщение Full LC содержит 72-битное информационное поле и передается в:

- речи и данных (встроено);
- заголовках;
- окончаниях.

Общая структура сообщения Full Link Control показана на рисунке 3.4.



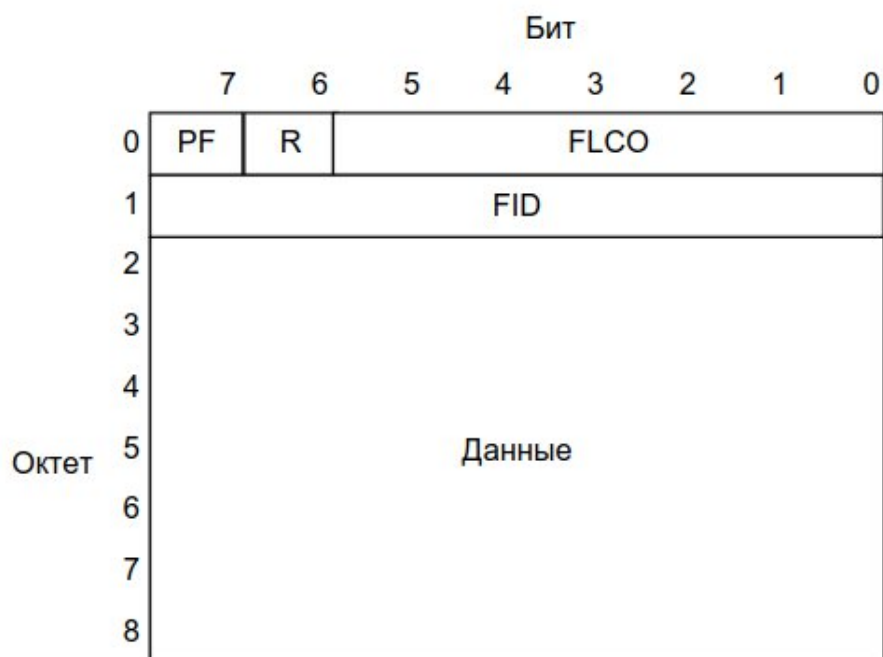


Рисунок 3.4 – Структура сообщения Full LC.

Full LC содержит 7 октетов данных связанных с кодом операций Full LC и комбинацией параметров ID (FID). Информационный элемент «Данные» содержит информации о специфических параметрах (например, Source ID и Destination ID).

Обобщенная структура элемента «Данные» приведена в таблице 3.1 [8].

Таблица 3.1 – Пакетные данные сообщения Full LC.

Информационный элемент	Длина	Примечание
Элементы зависящие от типа сообщения		
Protect Flag	1	
Reserved	1	Должен быть установлен 0 <sub>2</sub>
Характерные элементы		
Full Link Control Opcode	6	Зависит от типа сообщения Full LC

Feature set ID (FID)	8	Должен быть установлен $00000000_2$
Service Options	8	
Target address	24	
Source address	24	

При передаче «заголовка», пакет должен находиться в начале речевой передачи, использующей общий формат данных, обозначая начало передачи речи. На рисунке 3.5 показано, как 72-битное поле LC и 24-битный CRC передаются в едином общем пакете данных. Перед шифратором BPTC (196,96), должен быть применен соответствующий полином CRC типа данных, для получения 24 бит CRC. Биты «Тип данных» поля «Тип слота» должны задаваться как «Заголовок речевого LC» [7].

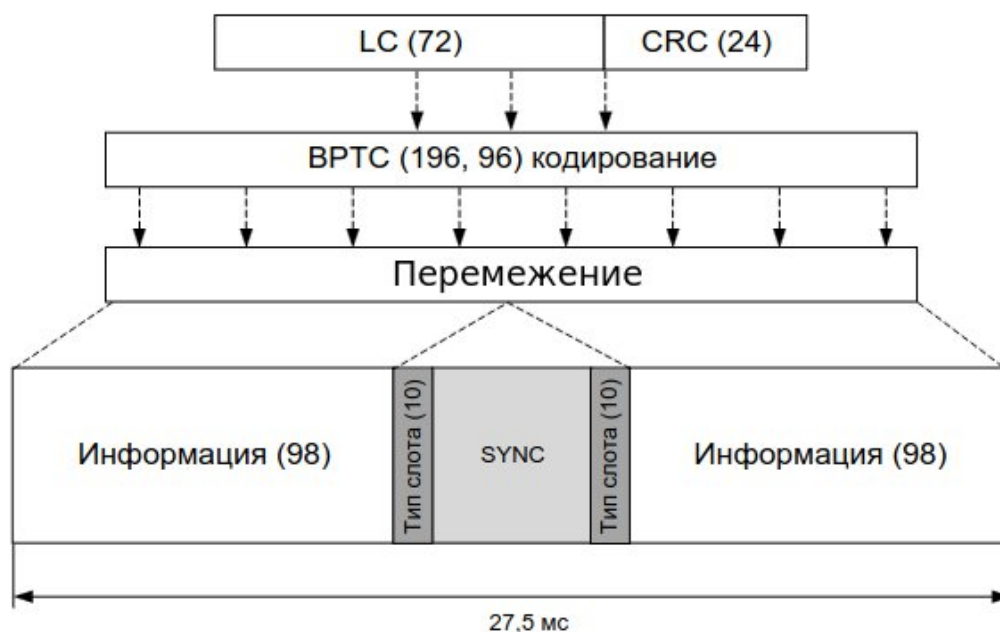


Рисунок 3.5 – Формирование речевого заголовка LC

Речевой вызов может быть завершен передачей пакета, который содержит синхрогруппу данных, сразу после передачи последнего речевого пакета. 72 бита информации о LC защищаются при помощи 24-битного CRC и FEC

BPTC, так же как и для заголовка изменяется лишь соответствующий полином для CRC и поле «Тип слота».

Чтобы обеспечить работу устройств, подключившихся к уже осуществляемой передаче, сообщения LC должны передаваться во встроенном поле речевых пакетов. 72-битное сообщение LC, после FEC кодирования и разделения на фрагменты, помещается во встроенное поле четырех пакетов. Это означает, что 6-пакетный речевой суперфрейм может выделить один пакет под SYNC, четыре пакета под LC, и один пакет под RC (как показано на рисунке 3.6) или встроенное Null сообщение (как показано на рисунке 3.7).

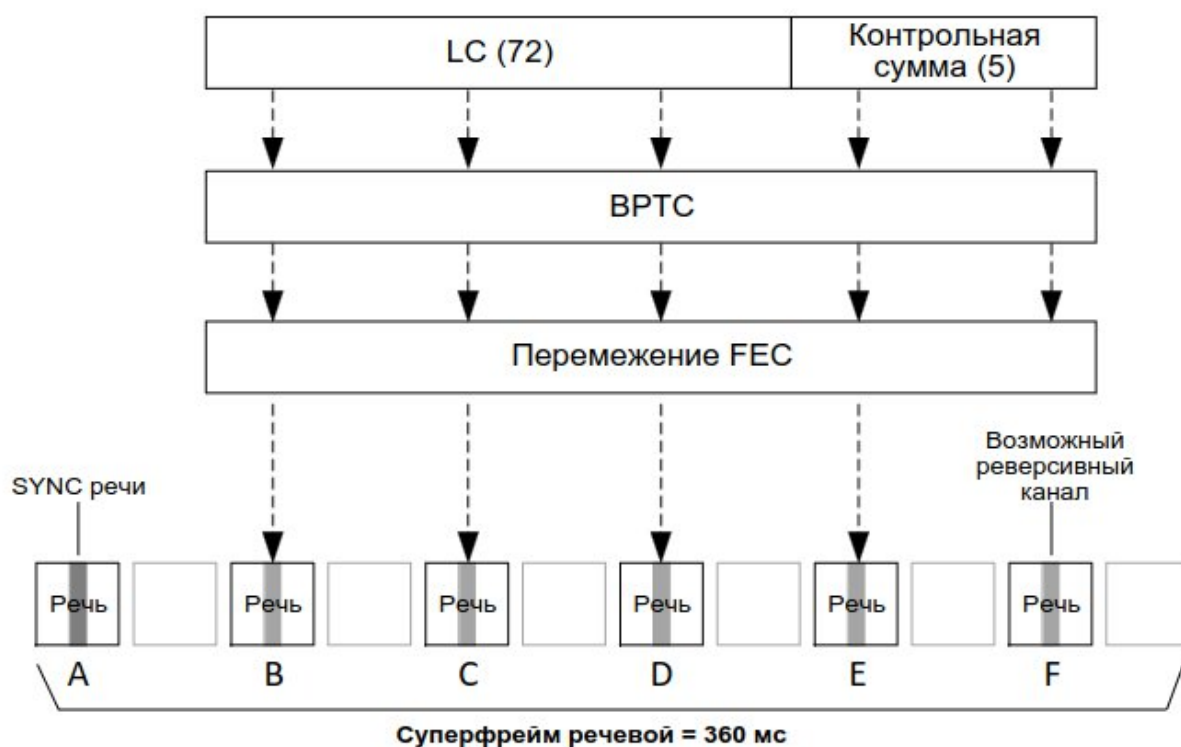


Рисунок 3.6 – Пример нисходящего голосового суперфрейма.

Все четыре пакета одного сообщения LC должны передаваться в одном речевом суперфрейме. Начало LC в первом не содержащем SYNC пакете (пакет В) суперфрейма. Начало, продолжение и конец полного сообщения LC структурируются, используя биты LCSS поля EMB. Типы встроенной сигнализации не LC обозначаются установкой бит поля LCSS, что обозначает содержание в них одного фрагмента пакета LC [7].

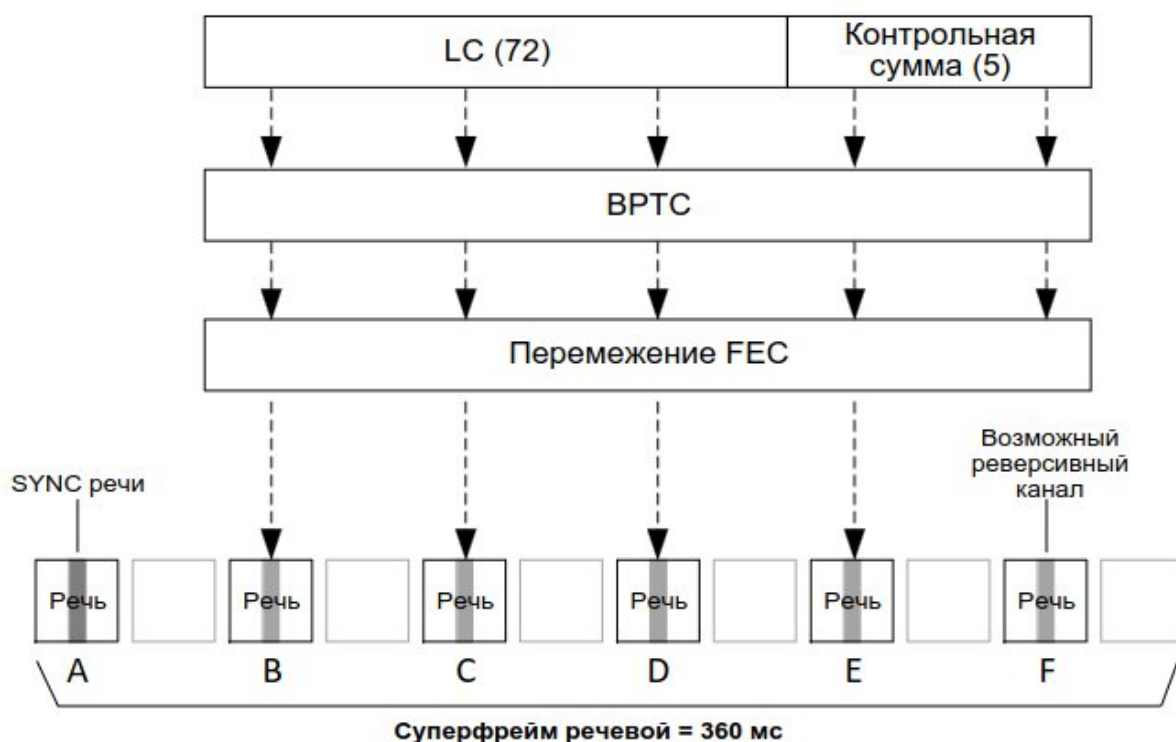


Рисунок 3.7 – Пример входящего голосового суперфрейма.

Пакет А всегда содержит речевую синхрогруппу, а пакет F – сигнализацию RC или ретранслирует восходящую информацию, связанную с конфиденциальностью или встроенное Null сообщение. Четыре речевых пакета, от В до Е, речевого суперфрейма содержат сообщение встроенной сигнализации. Например, на рисунке 3.6 показан исходящий голосовой суперфрейм, где пакет F является возможным RC каналом.

### Структура CSBK сообщений

Сообщение управляющего блока сигнализации содержит 96-битное информационное поле. Общая структура сообщения CSBK показана на рисунке 3.8.

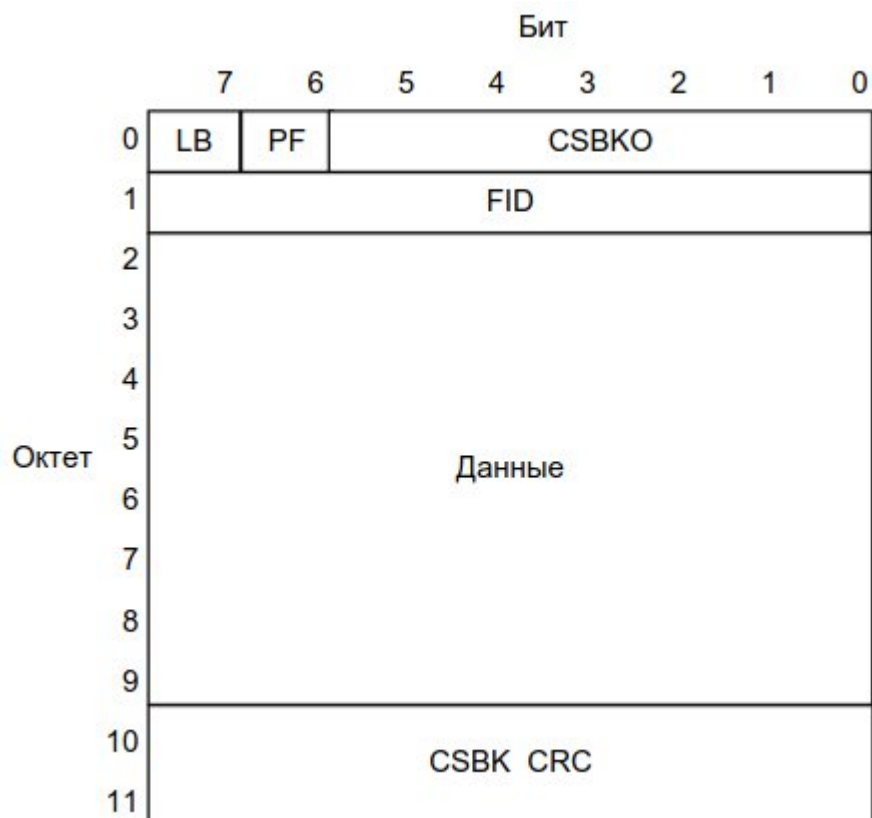


Рисунок 3.8 – Структура сообщения CSBK.

Сообщение CSBK содержит 8 октетов данных, связанных с кодом операции CSBK (CSBKO) и комбинацией ID набора функций (FID). Информационный элемент «последний блок» должен быть установлен в 1<sub>2</sub>. Информационный элемент «Данные» содержит характеристику данной информации (например, источник ID и адрес назначения ID).

Обобщенная структура элемента «Данные» приведена в таблице 3.2 [8].

Таблица 3.2 – Пакетные данные сообщения CSBK.

Информационный элемент	Длина	Примечание
Элементы зависящие от типа сообщения		
Last Block	1	
Protect Flag	1	

Характерные элементы		
CSBK Opcode (CSBKO)	6	Зависит от типа сообщения Full LC
Feature set ID (FID)	8	Должен быть установлен $00000000_2$
Service Options	8	
Reserved	8	Должен быть установлен $00000000_2$
Target address	24	
Source address	24	

Блок данных состоящий из 96 бит (80 бит сигнализации + 16 бит CRC) должен быть защищен FEC BPTC (196,96). Перед кодером BPTC (196,96), к 16 битам CRC должен быть применен соответствующий полином CRC типа данных. Информационные биты могут быть переданы в одном пакете данных, как показано на рисунке 3.9. Биты «Тип данных» поля «Тип слота» должны задаваться как CSBK [7].

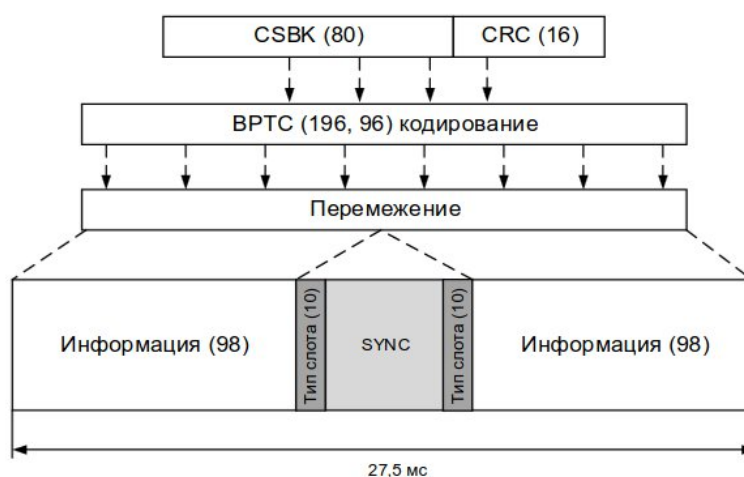


Рисунок 3.9 – Формирование сообщения CSBK.

### Структура Idle сообщений

Сообщение Idle передается BS, когда отсутствует сигнализация или трафик для отправки. Биты «Тип данных» поля «Тип слота» должны задаваться как «Idle». Информационные поля сообщений Idle будут заполнены заранее определенной псевдослучайной последовательностью (PR Fill).

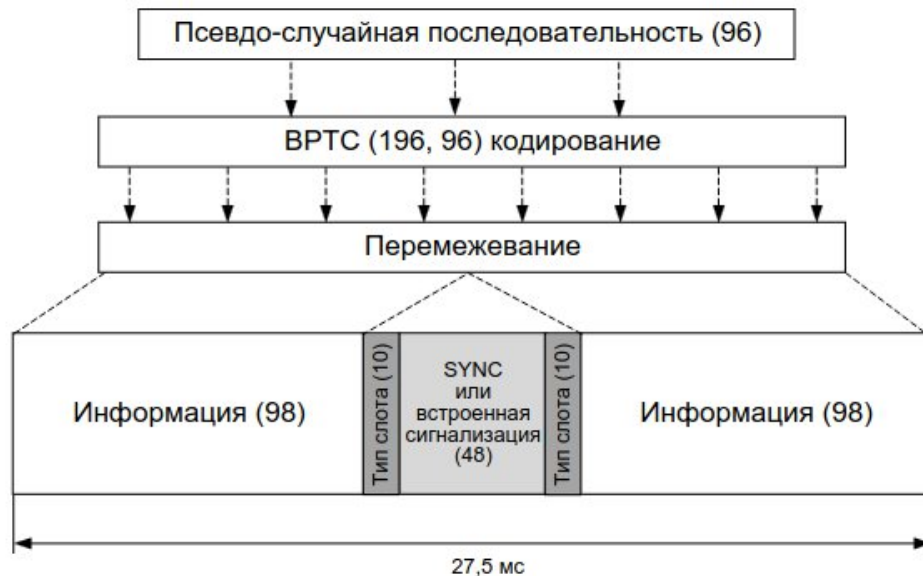


Рисунок 3.10 – Формирование сообщения Idle.

Эти биты кодируются с использованием FEC BPTC (196,96) и перемежения, используемого для обычных данных и управления, как показано на рисунке 3.10. Эти биты используются только для обеспечения непрерывной передачи BS. Они не предназначены для чтения или обработки MS [7].

### Структура MBS сообщений

Сообщение MBS используется в случае, если CSBK не может перенести всю необходимую управляющую информацию. Основной формат построен на структуре сообщения CSBK. Сообщение MBS должно состоять из заголовка MBS, промежуточных блоков MBS от 0 до 2 и последнего блока MBS. Общая структура трех различных блоков сообщения MBS показана на рисунках 3.11-3.13.

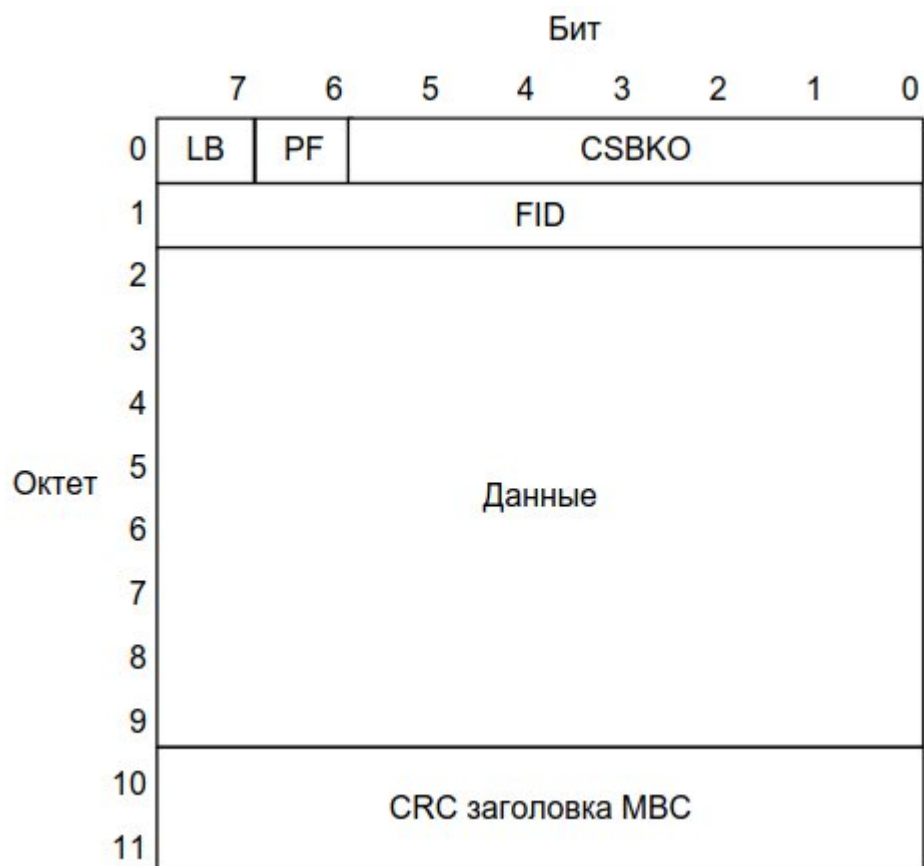


Рисунок 3.11 – Структура заголовка сообщения MBC.

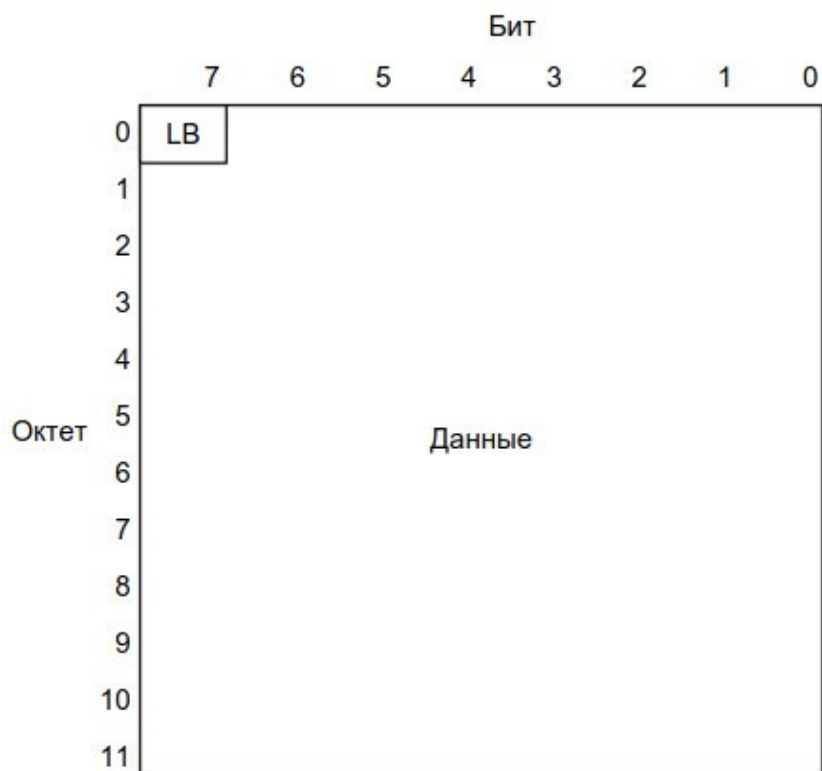


Рисунок 3.12 – Структура промежуточного блока сообщения MBC.



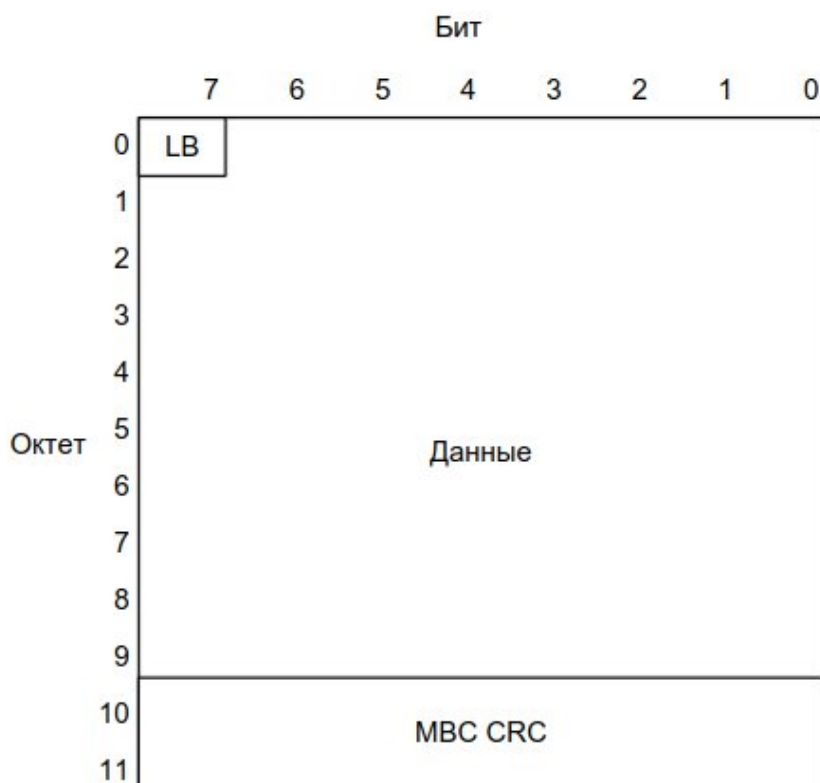


Рисунок 3.13 – Структура последнего блока сообщения MBC.

Сообщение MBC содержит данные, которые связаны с кодом операции CSBK (CSBKO) и комбинацией ID набора функций (FID). Информационный элемент «Данные» содержит характеристику конкретной информации и определен в ETSI TS 102 361-4. Блок заголовка содержит 64 бита, промежуточный блок – 95 бит, последний блок – 79 бит. Сообщение MBC несет до 333 бит данных, когда заголовок сопровождается 3 блоками. 16 бит CRC в заголовке должны быть включены в данные, передаваемые в заголовке. 16 бит CRC последнего блока должны быть получены по отношению к данным всех блоков MBC, кроме блока заголовка [7].

Заголовок MBC, промежуточные и последние блоки должны быть защищены FEC BPTC (196,96). Перед кодером BPTC (196,96) для заголовка MBC и последнего блока MBC должен быть применен соответствующий полином CRC типа данных для получения 16 бит CRC. Информационные биты при необходимости переносят в заголовке, промежуточных блоках и последнем блоке, как это показано на рисунках 3.14, 3.15 и 3.16. Биты «Тип дан-

ных» поля «Тип слота» должны задаваться как «Заголовок МВС» для заголовка и «продолжение МВС» для промежуточных и последнего блоков.

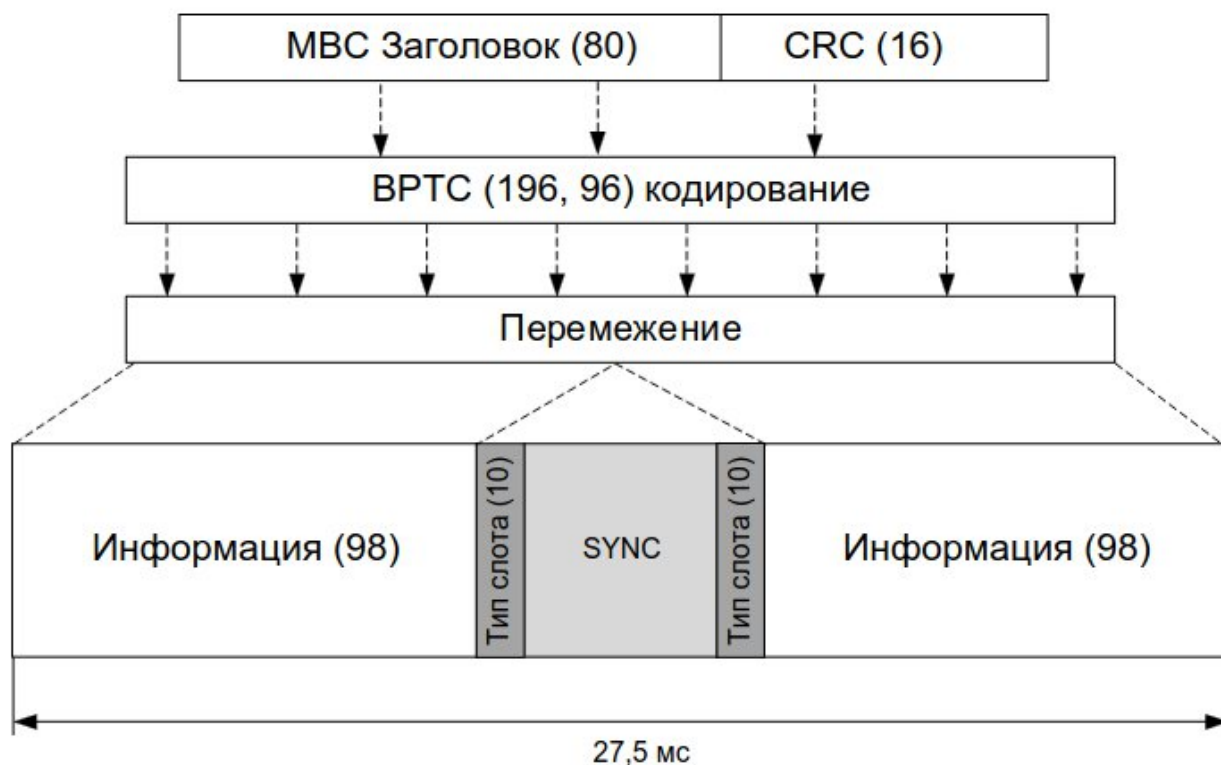


Рисунок 3.14 – Формирование заголовочного сообщения МВС.

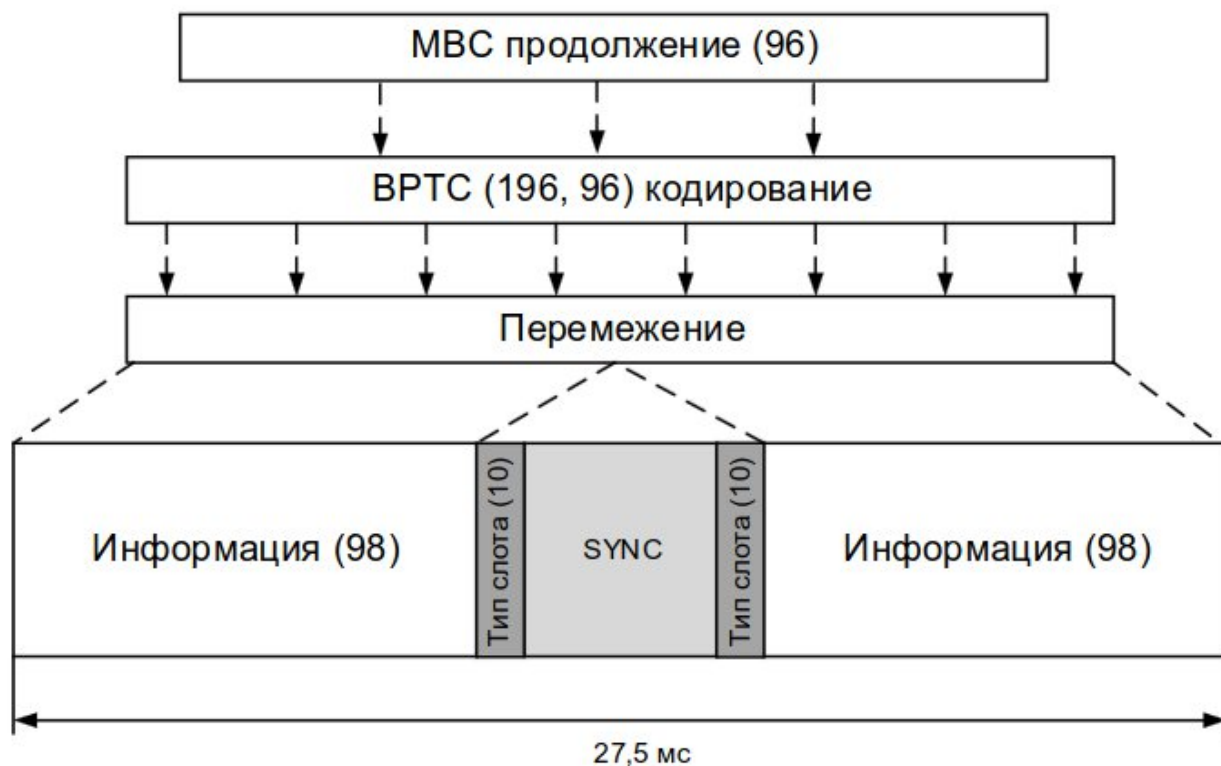


Рисунок 3.15 – Формирование промежуточного сообщения МВС.

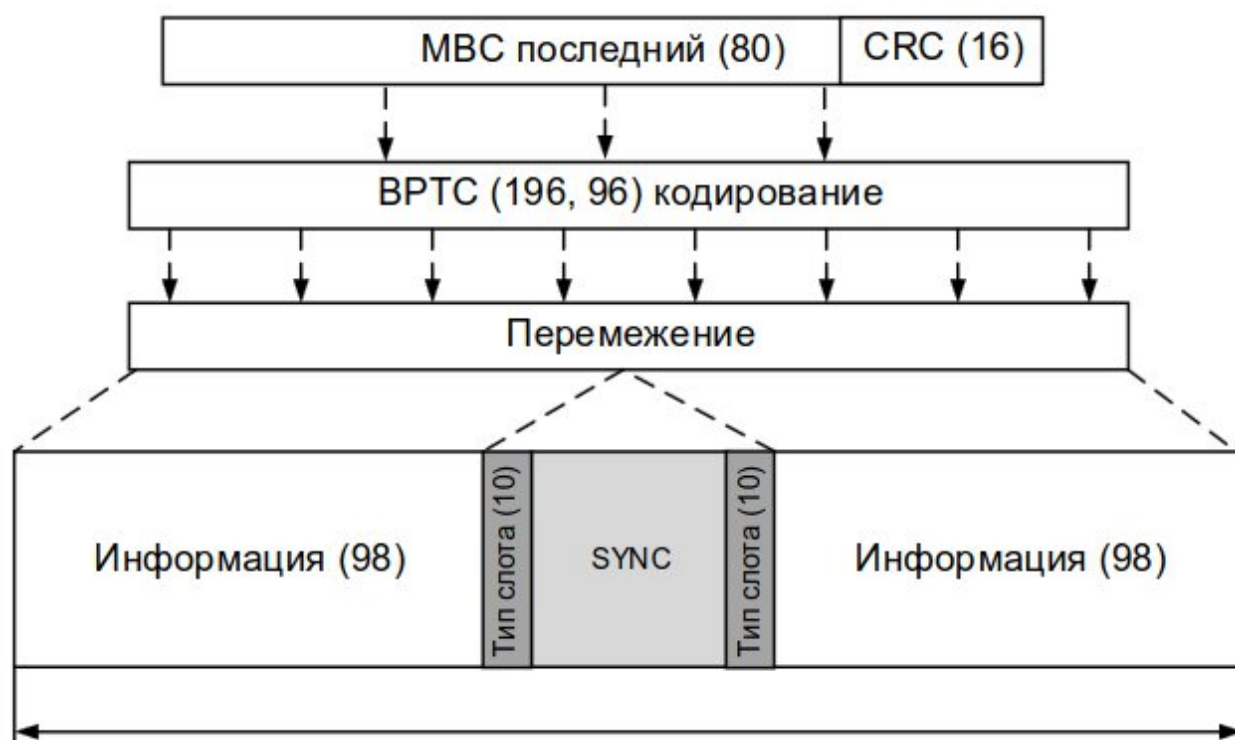


Рисунок 3.16 – Формирование последнего сообщения MBS.

### Тип слота

В каждом из описанных выше типов сообщений присутствует поле «Тип слота». Данное поле состоит из 20 бит и кодируется кодом Голея (20, 8). Состав поля «Тип слота» представлен в таблице 3.3 [7].

Таблица 3.3 – Состав поля «Тип слота».

Информационный элемент	Длина, бит
Цветовой код	4
Тип данных	4
Контрольные биты кода Голея	12

Информационный элемент «Цветовой код» позволяет различать сигналы исходящие из разных узлов. Информационный элемент «Тип данных» указывает тип данных или элемента управления, которые передаются в пакете данных, он может принимать значения из таблицы 3.4 [7].

Таблица 3.4 – Значения поля «Тип данных».

Значение	Примечание
0000 <sub>2</sub>	Privacy Indicator Header
0001 <sub>2</sub>	Voice LC Header
0010 <sub>2</sub>	Terminator with LC
0011 <sub>2</sub>	CSBK
0100 <sub>2</sub>	MBC Header
0101 <sub>2</sub>	MBC Continuation
0110 <sub>2</sub>	Data Header
0111 <sub>2</sub>	Rate ½ Data
1000 <sub>2</sub>	Rate ¾ Data
1001 <sub>2</sub>	Idle
1010 <sub>2</sub>	Rate 1 Data
1011 <sub>2</sub>	Unified Single Block Data
1100 <sub>2</sub> – 1111 <sub>2</sub>	Reserved for future use

### Синхронизация

Синхронизация кадров является начальным шагом к получению сообщения и должна быть выполнена до того, как встроенные поля могут быть извлечены, проанализированы и интерпретированы. Протокол TDMA состоит из режимов MS sourced voice, BS sourced voice, MS sourced data или control MS и BS sourced data или control BS, а также голосового интервала 1 прямого режима TDMA, данных временного интервала 1 прямого режима TDMA, голосового интервала 2 прямого режима TDMA и данных временного интервала 2 прямого режима TDMA.

Для различения этих режимов будут использоваться различные схемы синхронизации кадров. Используя начальную синхронизацию для переноса дополнительная информация, указывающая на эти режимы, уменьшает количество требуемых выделенных сигнальных битов в структуре пакета.

Протокол TDMA определяет уникальные 48-битные шаблоны синхронизации кадров для голоса и данных, они дополняют друг друга в виде символов. Коррелятор синхронизации кадров покажет положительный результат для голосового режима и такой же отрицательный результат для данных при запуске одного коррелятора.

DMR должен иметь синхрогруппы, приведенные в таблице 3.5.

Таблица 3.5 – Синхрогруппы.

BS sourced													
Voice	Hex	7	5	5	F	D	7	D	F	7	5	F	7
	Binary	0111	0101	0101	1111	1101	0111	1101	1111	0111	0101	1111	0111
Data	Hex	D	F	F	5	7	D	7	5	D	F	5	D
	Binary	1101	1111	1111	0101	0111	1101	0111	0101	1101	1111	0101	1101
MS sourced													
Voice	Hex	7	F	7	D	5	D	D	5	7	D	F	D
	Binary	0111	1111	0111	1101	0101	1101	1101	0101	0111	1101	1111	1101
Data	Hex	D	5	D	7	F	7	7	F	D	7	5	7
	Binary	1101	0101	1101	0111	1111	0111	0111	1111	1101	0111	0101	0111
RC sync	Hex	7	7	D	5	5	F	7	D	F	D	7	7
	Binary	0111	0111	1101	0101	0101	1111	0111	1101	1111	1101	0111	0111
TDMA direct mode time slot 1													
Voice	Hex	5	D	5	7	7	F	7	7	5	7	F	F
	Binary	0101	1101	0101	0111	0111	1111	0111	0111	0101	0111	1111	1111
Data	Hex	F	7	F	D	D	5	D	D	F	D	5	5
	Binary	1111	0111	1111	1101	1101	0101	1101	1101	1111	1101	0101	0101
TDMA direct mode time slot 2													
Voice	Hex	7	D	F	F	D	5	F	5	5	D	5	F
	Binary	0111	1101	1111	1111	1101	0101	1111	0101	0101	1101	0101	1111
Data	Hex	D	7	5	5	7	F	5	F	F	7	F	5
	Binary	1101	0111	0101	0101	0111	1111	0101	1111	1111	0111	1111	0101
Reserved SYNC pattern													
Reserved	Hex	D	D	7	F	F	5	D	7	5	7	D	D
	Binary	1101	1101	0111	1111	1111	0101	1101	0111	0101	0111	1101	1101

### 3.3. Физический уровень DMR

#### Общие параметры

Система радиосвязи DMR работает в радиочастотном диапазоне от 30 МГц до 1 ГГц. Ширина полосы пропускания сигнала составляет 12,5 кГц. Максимальная погрешность частоты передатчика BS на заданной центральной несущей РЧ представлена в таблице 3.5.

Таблица 3.5 – Погрешность частоты передатчика BS.

Частотный диапазон, МГц	Максимальная погрешность частоты BS, %
50 - 300	$\pm 0,0002$
300 - 600	$\pm 0,0001$
600 - 800	$\pm 0,000075$
800 - 1000	$\pm 0,0003$

Максимальная погрешность частоты передатчика MS на заданной центральной РЧ определена в таблице 3.6. В диапазоне от 600 МГц до 1 ГГц рекомендуется, чтобы частота MS была синхронизирована с BS, чтобы улучшить производительность системы.

Таблица 3.6 – Погрешность частоты передатчика MS.

Частотный диапазон, МГц	Максимальная погрешность частоты BS, %
50 - 600	$\pm 0,0002$
600 - 1000	$\pm 0,00015$

Максимальное отклонение частоты тактового генератора MS, не поддерживающей прямой режим TDMA, должно составлять  $\pm 0,0002\%$ . Такое отклонение частоты генератора, является приемлемым во время передачи MS. Перед передачей MS синхронизирует время с BS. Во время передачи MS допускается отклонением частоты тактового генератора в пределах максималь-

ного. Для MS, поддерживающей прямой режим TDMA, отклонение частоты генератора должно составлять  $\pm 0,0001\%$ . Эта величина отклонения частоты генератора приемлемая для поддержки работы канала MS в течение 10 минутного промежутка времени при предельных значениях отклонения частоты генератора. Производители должны быть осведомлены, что это отклонение частоты генератора определяется как температурной стабильностью, так и старением. Например, если температурная стабильность составляет  $\pm 0,00005\%$  для температуры от  $-30\text{ }^{\circ}\text{C}$  до  $+60\text{ }^{\circ}\text{C}$  и стабильность старения  $\pm 0,00005\%$  для 4 лет, это оборудование удовлетворяет требованиям отклонения частоты генератора для 4-летнего периода эксплуатации, при работе в диапазоне температур от  $-30\text{ }^{\circ}\text{C}$  до  $+60\text{ }^{\circ}\text{C}$ .

### Модуляция

Модуляция посылок – 4800 символов/с, с каждым символом передается 2 бита информации. Максимальное отклонение символа  $D$  определяется как:

$$D = 3h/2T$$

Где:

$h$  – индекс девиации, определенный для данной модуляции;

$T$  – длительность символа (1/4800) в секундах.

Индекс девиации  $h$  для 4FSK установлен 0,27. Этим определяется отклонение символа на 1,944 кГц от центра. Соответствие между символами и битами дано в таблице 3.7 [7].

Таблица 3.7 – Соответствие двухбитовых комбинаций символам девиации 4FSK.

Информационные биты		Символ	Девиация 4FSK
Бит 1	Бит 0		
0	1	+3	+1,944 кГц
0	0	+1	+0,648 кГц
1	0	-1	-0,648 кГц
1	1	-3	+1,944 кГц

Фильтр с характеристикой «приподнятый косинус» реализован для 4FSK таким образом, чтобы фильтр «приподнятый косинус» Найквиста использовался как для подавления внеполосных излучений передатчика, так и для подавления шума приемником. Вход размывающего фильтра для подавления внеполосных излучений передатчика получает серии импульсов, масштабируемых и разделенных во времени на 208,33 микросекунд (1/4800 с). Используемый метод разделения фильтром «приподнятого косинуса» Найквиста определяет частотную характеристику приемного фильтра – «квадратный корень из приподнятого косинуса» Найквиста. Групповая задержка фильтра является плоской по полосе пропускания для  $|f| < 2880$  Гц. АЧХ фильтра задается приблизительно следующей формулой:

$$|F(f)| = 1 \text{ для } |f| \leq 1920 \text{ Гц}$$

$$|F(f)| = |\cos(\pi f/1920)| \text{ для } 1920 \text{ Гц} < |f| \leq 2880 \text{ Гц}$$

$$|F(f)| = 0 \text{ для } |f| > 2880 \text{ Гц}$$

где  $F(f)$  – АЧХ фильтра «квадратный корень из приподнятого косинуса».

4FSK модулятор состоит из фильтра «квадратного корня приподнятого косинуса», каскадно включенного с частотным модулятором как показано в рисунке 3.17.

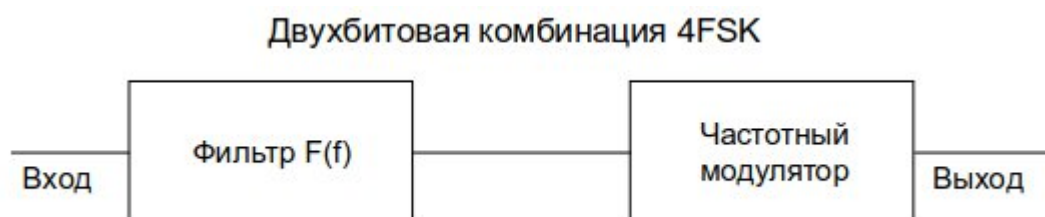


Рисунок 3.17 – 4FSK модулятор.

4FSK модулятору необходимо установить отклонение для обеспечения надлежащего сдвига фаз несущей для каждого модулируемого символа. Отклонение устанавливается по тестовому сигналу, состоящему из следующего потока символов:



$$\dots +3 +3 - 3 - 3 +3 +3 - 3 - 3 \dots$$

Этот тестовый сигнал обрабатывается модулятором для создания сигнала 4FSK, эквивалентного синусоидальной волне 1,2 кГц модулированной сигналом FM с пиковым отклонением, равным:

$$\sqrt{2} * 1944 \text{ Гц} = 2749 \text{ Гц}$$

Демодулятор FM используется, чтобы измерить как положительный пик, так и отрицательный пик девиации. Полоса частот аудиосигнала демодулятора FM устанавливается с пороговой частотой фильтра высоких частот  $\leq 15 \text{ Гц}$  и пороговой частотой фильтра низких частот  $\geq 3 \text{ кГц}$ .

### Временные параметры пакетов

Передачи в системе TDMA включают короткие пакеты и постоянные интервалы. Временные параметры этих пакетов важны для эффективного функционирования системы TDMA. Существует два типа пакетов, определенных для протокола:

- обычные пакеты;
- пакеты RC.

Оба используют фрейм TDMA и структуру слотов, показанную на рисунке 3.18.

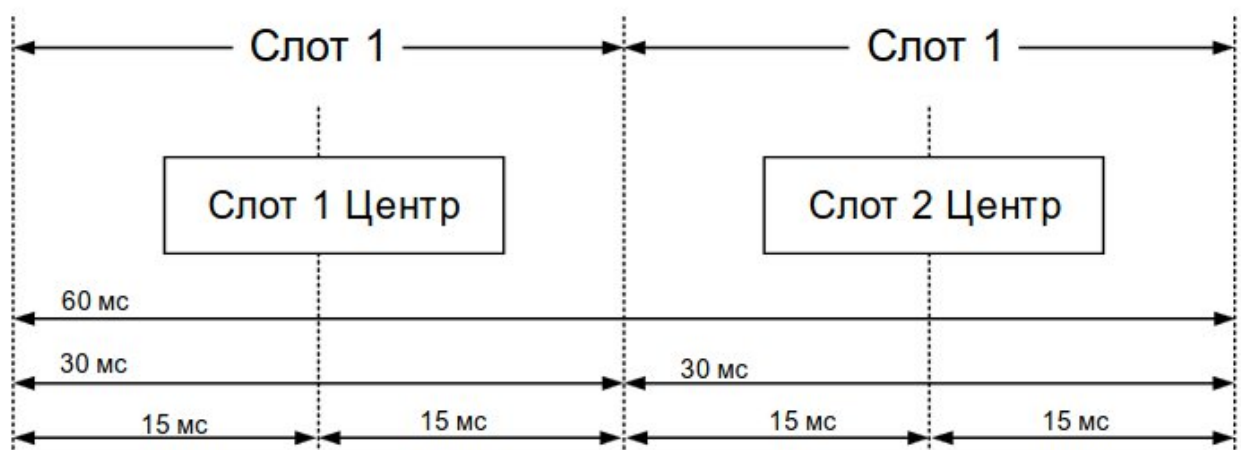


Рисунок 3.18 – Фрейм TDMA.

Каждый кадр TDMA длиной 60 мс и состоит из двух таймслотов по 30 мс. Обычно первый вызов использует слот 1, а другой вызов использует слот

2. Вызовы состоят из серии равных временных слотов. Для систем, использующих BS – мобильный терминал синхронизируется от БС. Информация, которая передается во временном слоте, размещается в центре слота. Обычный пакет должен использоваться для приложений речи, данных и управления. Он предусматривает 264 бита данных на пакет со скоростью передачи данных 4,4 Кбит/с. Это пакет, используется для большинства приложений.

### **Перемежение**

*Блочный турбокод Block Product Turbo Codes BPTC (196,96).*

Передача сигналов управления, неподтверждаемых и подтверждаемых данных защищенных с применением 196,96 блоков, являющихся результатом турбокодирования, показана на рисунке 3.19 [7], 96 бит информации  $I(0) - I(95)$  размещены, как показано, в 9 строках и 11 столбцах матрицы. Три зарезервированных бита,  $R(0) - R(2)$ , установлены в ноль и добавлены для округления полезной нагрузки до 99 бит. Каждая строка защищена с помощью кода Хэмминга (15,11,3), обозначенными битами  $H_{Rx}$ . Каждый столбец защищен с помощью кода Хэмминга (13,9,3) обозначенными битами  $H_{Cx}$ .

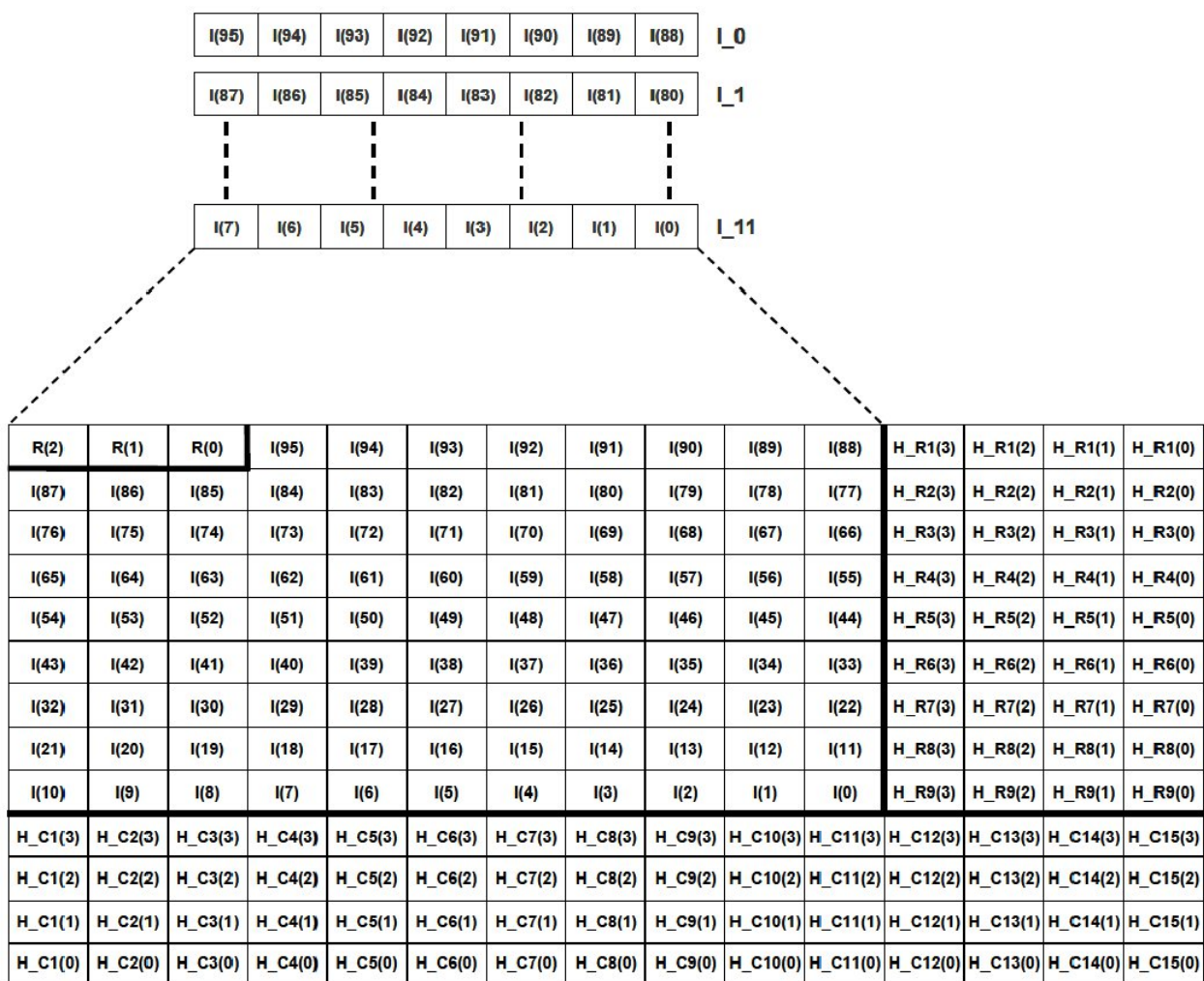


Рисунок 3.19 – ВРТС (196,96).

Первым шагом в перемежении битов для передачи необходимо последовательно пронумеровать биты закодированной матрицы FEC сначала слева направо и затем сверху донизу. Чтобы увеличить общее количество битов до 196, один дополнительный зарезервированный бит  $R(3)$ , устанавливается в ноль и добавляется к матрице и ему присваивается индекс 0. Каждому биту тогда присваивается новый индекс в массиве перемежения, где:

$$\text{Индекс перемежения} = \text{Индекс} \times 181 \text{ по модулю } 196$$

Значение индекса перемежения определяет местоположение каждого бита в массиве передачи, помещенном в полезную нагрузку общего пакета данных. Значения индекса от 0 до 195, соответствуют индексу перемежения из предыдущей таблицы. Полученный массив содержит 195 битов, пронуме-

рованных от ТХ (195) вниз к ТХ (0) для размещения в полезной нагрузке общего пакета данных.

### Кодирование

*Код Голя (20,8)*

(20,8,7) код Голя получается путем сокращения примитивного кода, произведенного от полинома  $g(x)$  заданного ниже:

$$g(x) = x^{11} + x^{10} + x^6 + x^5 + x^4 + x^2 + 1 = 6165_8$$

Порождающая матрица дана в таблице 3.8 [7].

Таблица 3.8 – Порождающая матрица кода Голя (20,8).

1	0	0	0	0	0	0	0	0	0	1	1	1
0	1	0	0	0	0	0	0	1	1	0	1	1
0	0	1	0	0	0	0	0	0	1	1	0	1
0	0	0	1	0	0	0	0	0	0	1	1	0
0	0	0	0	1	0	0	0	1	1	0	1	1
0	0	0	0	0	1	0	0	1	0	1	0	1
0	0	0	0	0	0	1	0	1	0	0	1	0
0	0	0	0	0	0	0	1	1	0	0	0	1

*Код Хэмминга (13,9,3) и (15,11,3)*

Порождающая матрица для кода Хемминга (13,9,3) получена из примитивного кода Хемминга (15,11,3). Порождающий полином для примитивного кода следующий:

$$G(x) = x^4 + x + 1 = 23_8$$

Порождающая матрица дана в таблицах 3.9 и 3.10 [7].

Таблица 3.9 – Порождающая матрица кода Хэмминга (13,9,3).

1	0	0	0	0	0	0	0	0	1	1	1	1
0	1	0	0	0	0	0	0	0	1	1	1	0
0	0	1	0	0	0	0	0	0	0	1	1	1
0	0	0	1	0	0	0	0	0	1	0	1	0

Продолжение таблицы 3.9

0	0	0	0	1	0	0	0	0	0	1	0	1
0	0	0	0	0	1	0	0	0	1	0	1	1
0	0	0	0	0	0	1	0	0	1	1	0	0
0	0	0	0	0	0	0	1	0	0	1	1	0
0	0	0	0	0	0	0	0	1	0	0	1	1

Таблица 3.10 – Порождающая матрица кода Хэмминга (13,9,3).

1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1
0	1	0	0	0	0	0	0	0	0	0	1	1	0	1	0
0	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1
0	0	0	1	0	0	0	0	0	0	0	1	1	1	0	0
0	0	0	0	1	0	0	0	0	0	0	0	1	1	1	0
0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	1
0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1
0	0	0	0	0	0	0	1	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	1
0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	1
0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1

## 4. АЛГОРИТМ ОБРАБОТКИ СИГНАЛА

В этой главе описывается алгоритм обработки сигнала стандарта DMR, его основные этапы, реализация и результаты работы алгоритма.

### 4.1. Устройство АРМ

В ходе выполнения ВКР основной целью является реализация программного модуля обработки сигнала стандарта DMR, программный модуль является частью ПО, используемого устройством АРМ. На рисунке 4.1 приведена обобщенная структурная схема устройства АРМ.

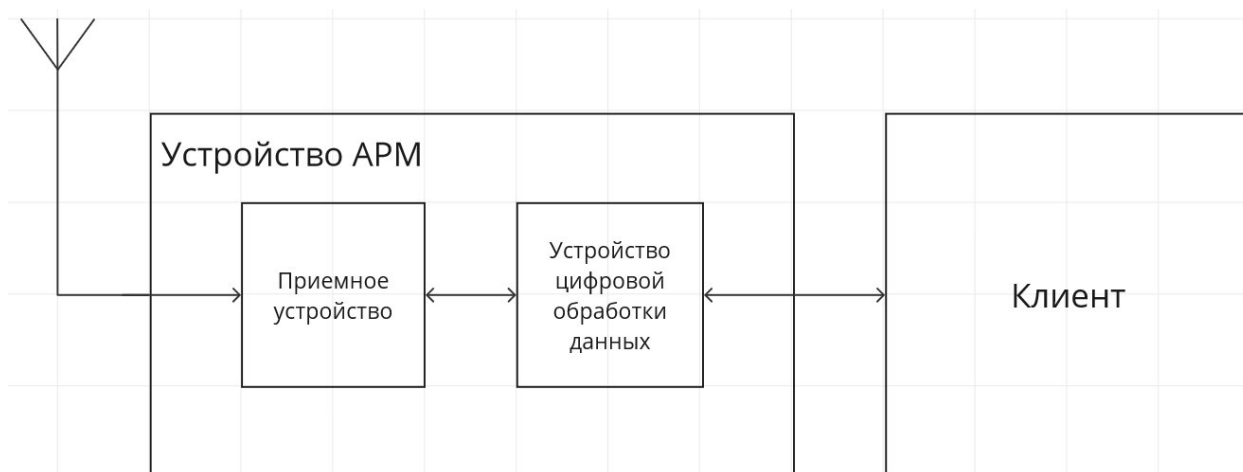


Рисунок 4.1 – Обобщенная структурная схема устройства АРМ.

Приемное устройство (ПУ) – в данной задаче включает в себя весь приемный тракт от антенны до выхода аналого-цифрового преобразователя. На выходе ПУ – цифровой сигнал, принимаемый с некоторой несущей частоты и переносимый на промежуточную.

Устройство цифровой обработки данных (УЦОД) – устройство, выполняющее обработку запросов, заданий, формирование ответов и т. д., – является основным звеном и выполняет почти всю цифровую обработку сигнала. В УЦОД задействовано ПО, компонентом которого является программный модуль, реализованный в ВКР.

Клиент – этот модуль фактически не присутствует в устройстве, но он показан на схеме для иллюстрации работы алгоритма. Клиент напрямую связан с пользователем. Задания отправляются с клиента, на него же отправляются отчеты о ходе выполнения заданий и сообщения об ошибках.

Взаимодействие между описанными блоками осуществляется следующим образом:

- Клиент отправляет задание, включающее в себя диапазон сканирования, шаг сетки частот, длительность сканирования одной частоты и режим сканирования (непрерывный или одиночный), устройству цифровой обработки данных, оно в свою очередь формирует массив запрос к ПУ.
- Далее ПУ отправляет в УЦОД блоки данных, содержащие массивы синфазных ( $I$ ) и квадратурных ( $Q$ ) компонент, частоту, с которой производилась запись и номер ответа.
- Получив ответ от ПУ, УЦОД осуществляет, непосредственно обработку (обнаружение сигнала, синхронизацию и т. д.). После окончания обработки, УЦОД отправляет полученные результаты клиенту.

Далее будут рассмотрены основные этапы алгоритма обработки данных в УЦОД, реализованные в ходе выполнения выпускной квалификационной работы.

Входными данными для алгоритма является запись сигнала, полученная от ПУ, которая представляет собой комплексную огибающую аналитического сигнала, она представлена массивом комплексных отсчетов.

Выходными данными должны быть статус декодирования и системная информация об устройстве.

## **4.2. Основные этапы алгоритма**

Основываясь на принципах построения сигнала стандарта DMR, предложен и реализован следующий алгоритм обработки сигнала:

- Первым этапом является коррекция частотной ошибки, вызванной эффектом Доплера и неточностью установки центральной частоты ПУ. Коррекция осуществляется с использованием класса *FreqErrorCompensator*. Данный класс является частью коммерческой библиотеки ООО «СТЦ». Входными и выходными данными являются комплексные отсчеты сигнала;
- Вторым этапом является обнаружение синхрогруппы сигнала DMR в потоке данных. Обнаружение осуществляется с использованием класса *DmrPreamble*. Входными данными являются комплексные отсчеты сигнала, выходными – массив позиций синхрогруппы в наборе входных данных. Алгоритм обнаружения был разработан мной, его описание приведено в разделе 4.4;
- Третьим этапом является демодуляция сигнала. Демодуляция осуществляется с использованием класса *DmrDemodulator*. Входными данными являются комплексные отсчеты сигнала, выходными – набор демодулированных символов. Алгоритм демодуляции был разработан мной, его описание приведено в разделе 4.3;
- Четвертым этапом является декодирование системной информации об устройстве. Декодирование осуществляется с использованием класса *DmrDeviceInfoExtractor*. Входными данными является набор демодулированных символов и позиции начала кадра, выходными – массив, состоящий из пар: позиция начала кадра и системная информация об устройстве. Алгоритм декодирования был разработан мной, его описание приведено в разделе 4.5;
- Последним этапом алгоритма является формирование результата работы алгоритма. При формировании результата учитывается был ли обнаружен сигнал и был ли обнаруженный сигнал декодирован или нет. В случае обнаружения сигнала формируется пара значений: позиция начала кадра в полученном сигнале и тип обнаруженной синхро-



группы. В случае декодирования к уже имеющейся структуре добавляется системная информация.

Описанный алгоритм был реализован как метод *process()* класса *DmrCore*, содержание данного класса является коммерческой тайной ООО «СТЦ». Результатом работы метода являются два возможных состояния:

- *detected* (обнаружено) – получение данного состояния возможно только в случае, когда сигнал был обнаружен. В этом случае для получения результатов обнаружения и декодирования необходимо вызвать метод *getSearchRes()*.
- *notDetected* (не обнаружено) – получение данного состояния возможно в случае, когда сигнал не был обнаружен.

Используемые в реализации алгоритма, классы *DmrCore*, *DmrPreamble*, *DmrDemodulator*, и *DmrDeviceInfoExtractor* являются моей разработкой, далее будут рассмотрены алгоритмы их работы. На рисунке 4.2 изображена блок-схема алгоритма. На рисунке 4.3 приведена UML-диаграмма отношений этих классов. На UML-диаграмме представлены перечисленные классы, их взаимоотношения и публичные методы используемые этими классами.



Рисунок 4.2 – Блок-схема алгоритма обработки сигнала.

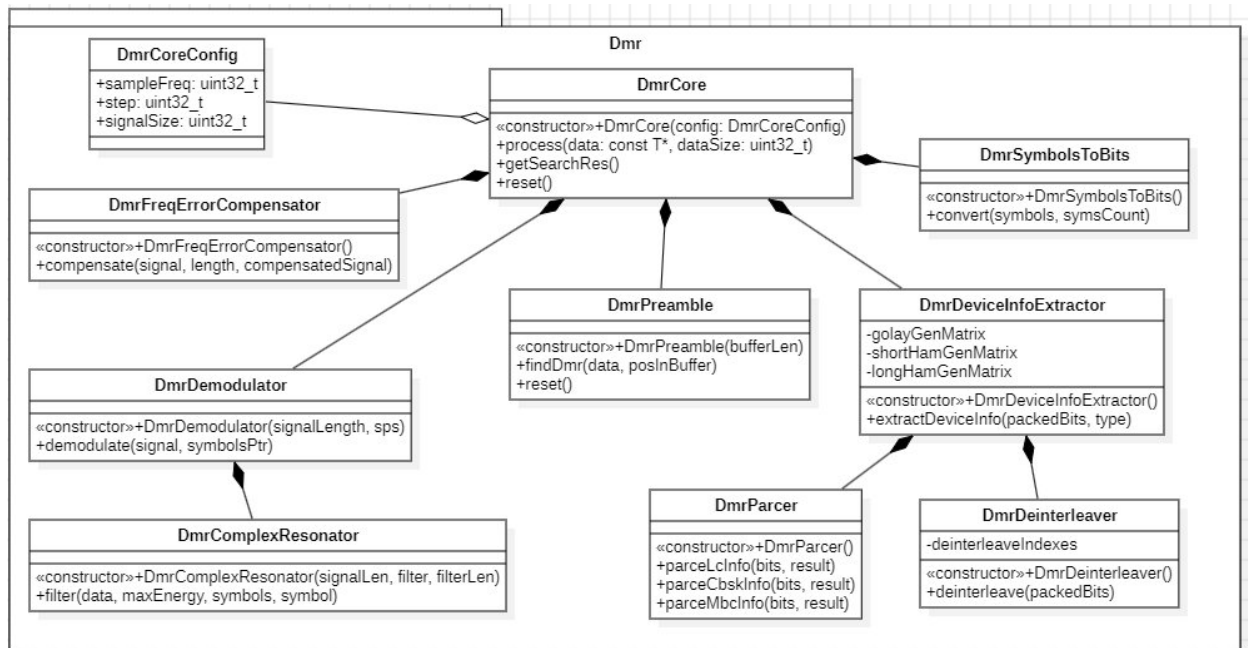


Рисунок 4.3 – UML-диаграмма отношений классов.

### 4.3. Демодуляция

Целью демодуляции является получение оценки символов. Алгоритм демодуляции предполагает использование четырех полосовых фильтров с центральными частотами, которые соответствуют модуляции 4FSK, используемой в стандарте DMR. АЧХ фильтров представлена на рисунке 4.4.

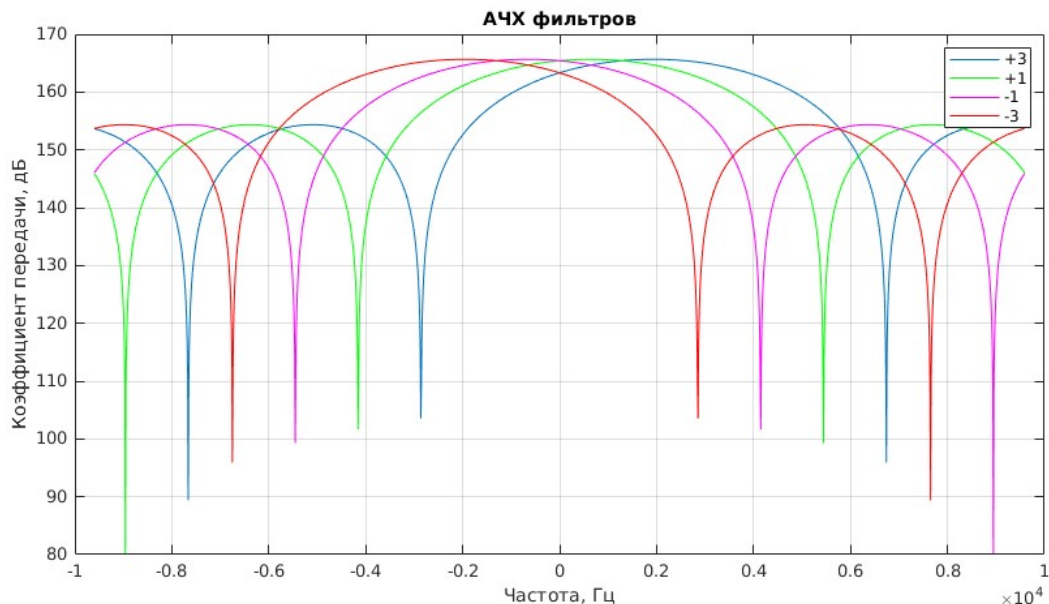


Рисунок 4.4 – АЧХ фильтров демодулятора.

Алгоритм демодуляции состоит из следующих этапов:

- расчет ВКФ принятого сигнала  $I(i), Q(i)$  с импульсной характеристикой фильтра;
- расчет энергии  $E(i)$  каждого отсчета ВКФ, как квадрата модуля этого отсчета;
- сравнение значения энергии для каждого отсчета ВКФ с предыдущими значениями, перед расчетом ВКФ с первым фильтром, значения энергии равны нулю. Если значение энергии для текущего отсчета превышает уже имеющееся, в массив демодулированных символов  $S(i)$  записывается символ соответствующий текущему фильтру.

Блок-схема описанного алгоритма демодуляции представлена на рисунке 4.5.

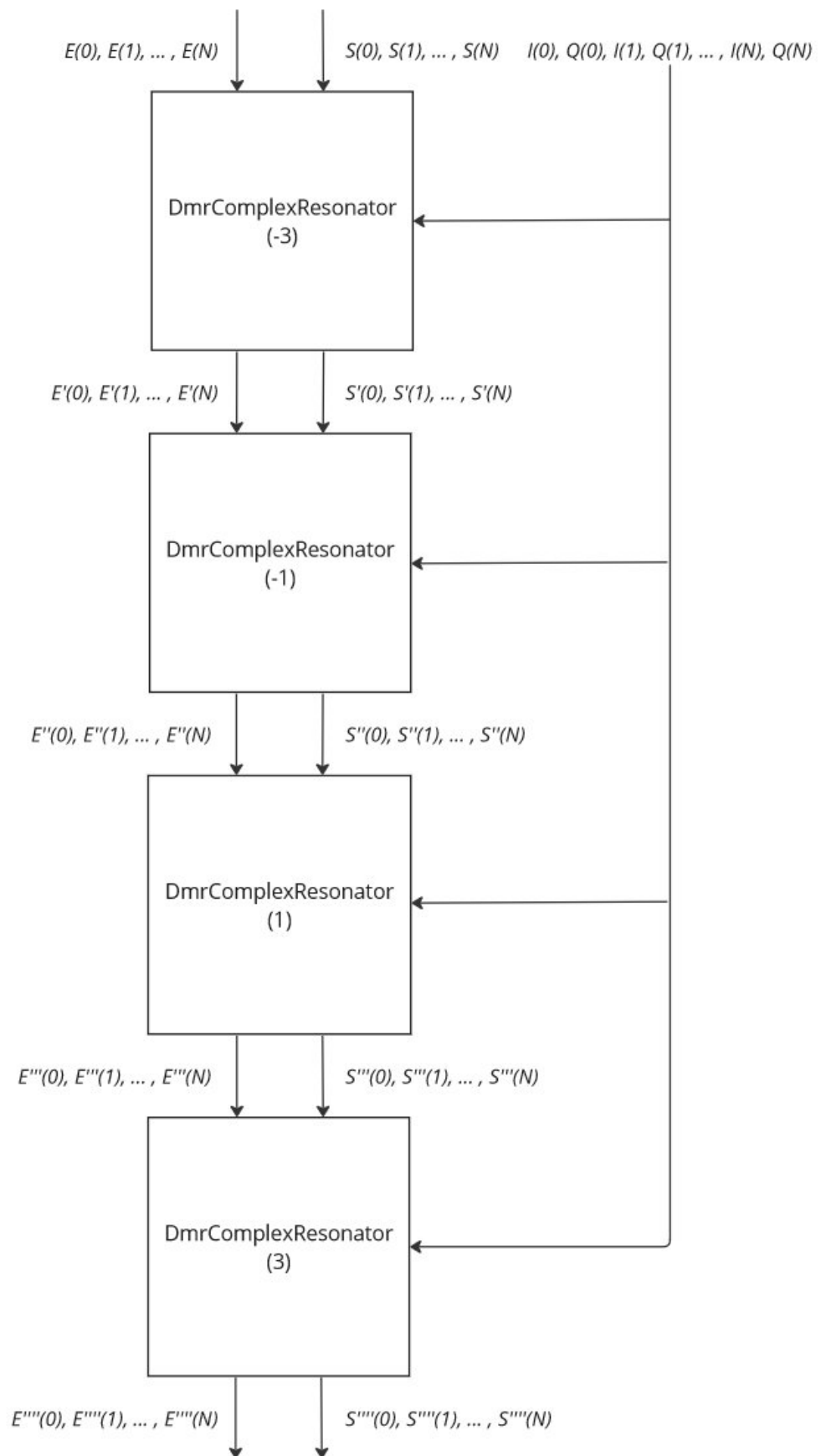


Рисунок 4.5 – Блок-схема алгоритма демодуляции 4FSK.

Описанный алгоритм реализован как метод *demodulate()* класса *DmrDemodulator*. Класс *DmrDemodulator* содержит в себе четыре вектора комплексных отсчетов, представляющих собой импульсные характеристики фильтров и вектор символов. Реализация класса *DmrDemodulator* является коммерческой тайной ООО «СТЦ».

Верификация алгоритма демодуляции 4FSK проводилась путем сравнения символов, полученных после демодуляции с соответствующими символами тестового сигнала. Для генерации тестового сигнала была использована функция *fskmod()* из ПО MATLAB, с параметрами соответствующими стандарту DMR, данная функция генерировала исключительно тестовый сигнал без шума. Аналогичные тестовые сигналы использовались при верификации кода на C++, в результате были получены аналогичные выходные данные.

Помимо описанного алгоритма демодуляции 4FSK, в ходе разработке алгоритма для исследования обнаружения, мной был реализован алгоритм демодуляции FM сигнала. Основой алгоритма демодуляции FM является следующая формула [9]:

$$x_{FM}(t) = \frac{\frac{dI}{dt} * Q - \frac{dQ}{dt} * I}{Q^2 + I^2},$$

где  $x_{FM}(t)$  – частотная функция принятого сигнала,  $I$  – синфазная составляющая принятого сигнала,  $Q$  – квадратурная составляющая принятого сигнала. Поскольку принятый сигнал представлен дискретными отсчетами, производная по времени рассчитана как разность следующего и предыдущего отсчетов.

Верификация алгоритма демодуляции FM проводилась путем демодуляции ЛЧМ сигнала, сгенерированного мной в MATLAB с использованием функции *modulate()*, и последующего сравнения результата на выходе демодулятора с исходным модулирующим сигналом.

#### 4.4. Обнаружение

Изначально для реализации был предложен алгоритм обнаружения по демодулированным символам. Поскольку отчасти данный алгоритм уже был реализован в ООО «СТЦ», было предложено провести сравнительный анализ этого алгоритма с алгоритмами обнаружения по комплексным отсчетам и по демодулированным отсчетам частоты.

На рисунках 4.6-8 приведены блок-схемы предложенных алгоритмов обнаружения. Опорный сигнал коррелятора  $Y_1(t)$  представляет собой набор символов синхрогрупп описанных в таблице 3.5 и преобразованных из бит в символы согласно таблице 3.7. Опорный сигнал детектора огибающей  $Y_2(t)$  представляет собой комплексный сигнал полученный путем модуляции сигнала  $Y_1(t)$  с помощью функции *fskmod()* из ПО MATLAB. Опорный сигнал коррелятора  $Y_3(t)$  представляет собой результат FM демодуляции сигнала  $Y_2(t)$ .

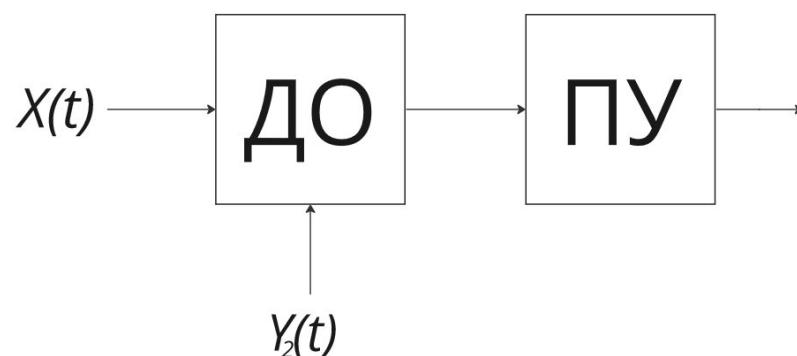


Рисунок 4.6 – Блок-схема алгоритма обнаружения по комплексным отсчетам.

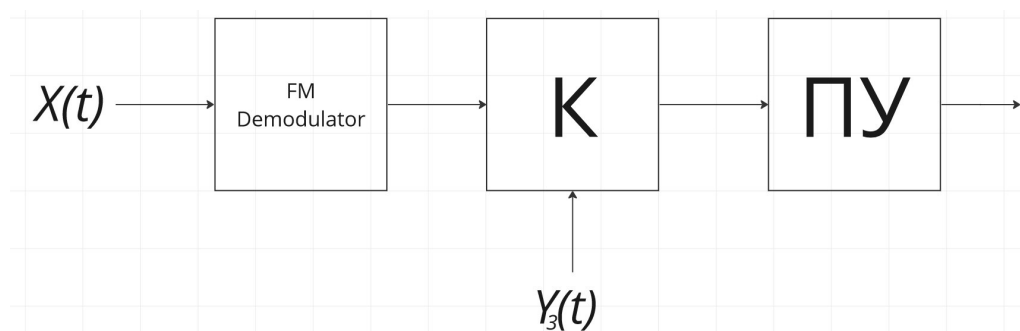


Рисунок 4.7 – Блок-схема алгоритма обнаружения по демодулированным отсчетам частоты.

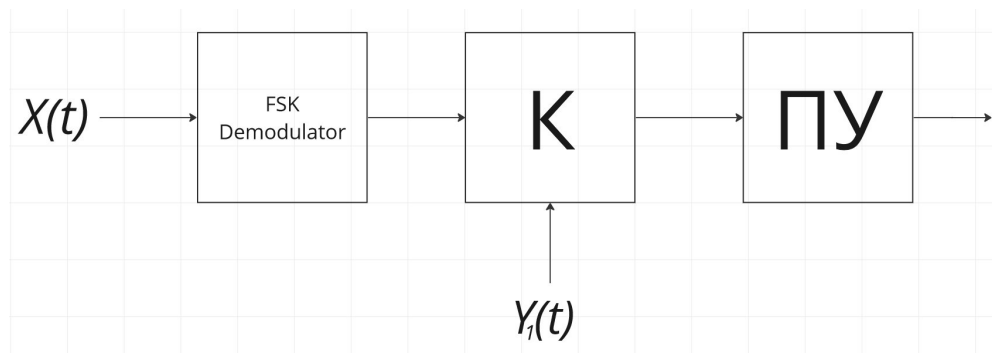


Рисунок 4.8 – Блок-схема алгоритма обнаружения по демодулированным символам.

Для верификации алгоритма обнаружения в MATLAB был сгенерирован тестовые сигналы содержащие сигналы всех синхрогрупп. Далее каждый сигнал был подан на вход обнаружителя, в результате все сигналы были успешно обнаружены.

Для выбора алгоритма было проведено исследование характеристик обнаружения и быстродействия каждого из предложенных вариантов.

Для расчета порога обнаружения для каждого из представленных алгоритмов был выбран критерий Неймана-Пирсона, так как априорная вероятность наличия или отсутствия устройств в заданной области пространства и потери из-за ложной тревоги или пропуска целей неизвестны. Согласно представленным требованиям к программному модулю, максимальная вероятность ложной тревоги для алгоритма обнаружения не должна превышать  $10^{-4}$ .

Моделью для выбора порога обнаружения служит выборка, состоящая из  $10^6$  независимых максимумов ВКФ сигнала синхропоследовательности  $Y_i(t)$  с белым гауссовским шумом  $X(t)$ .

На рисунке 4.9 приведена гистограмма белого гауссовского шума на входе схем, на рисунках 4.10-12 приведены гистограммы на входе порогового устройства.

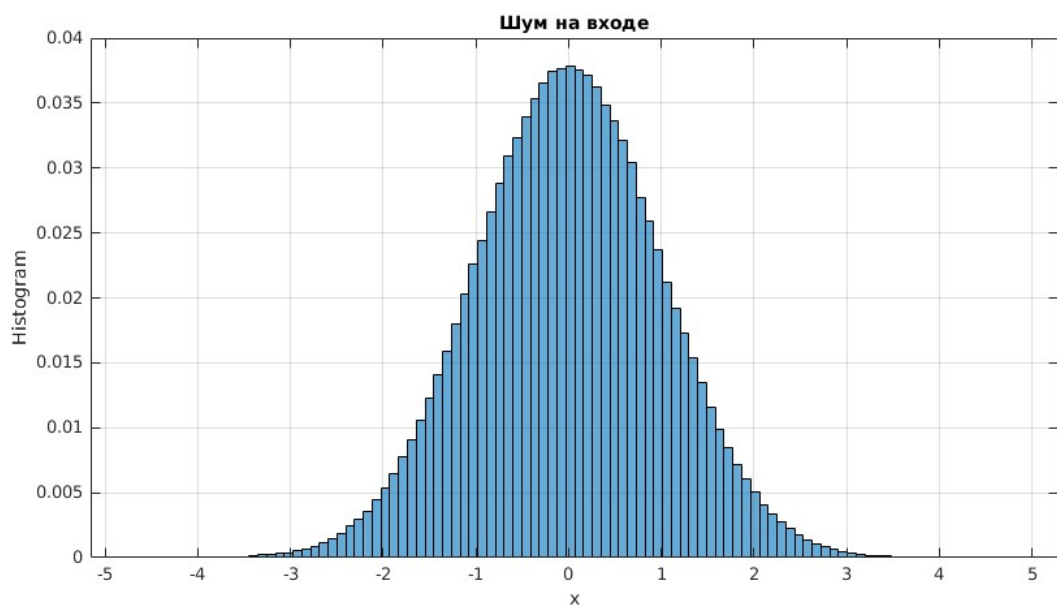


Рисунок 4.9 – Гистограмма шума на входе.

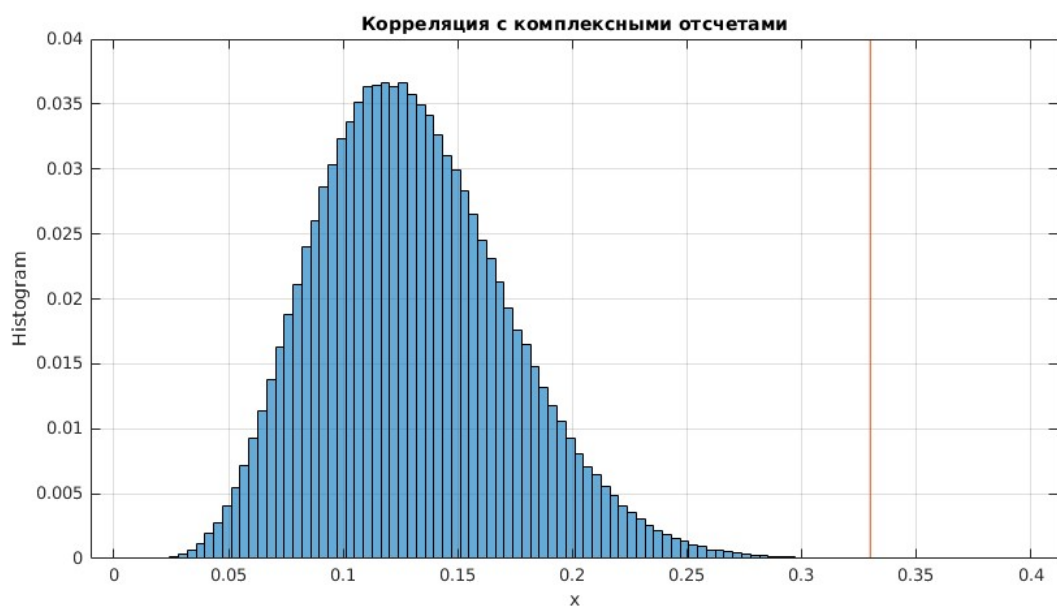


Рисунок 4.10 – Гистограмма максимумов корреляции с шумом по комплексным отсчетам.



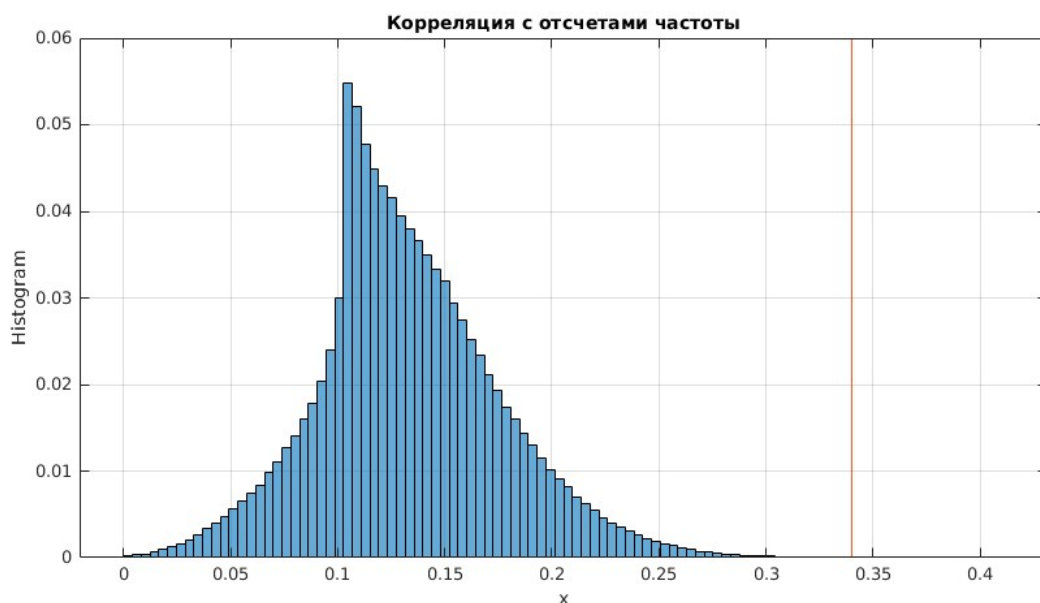


Рисунок 4.11 – Гистограмма максимумов корреляции с шумом по демодулированным отсчетам частоты.

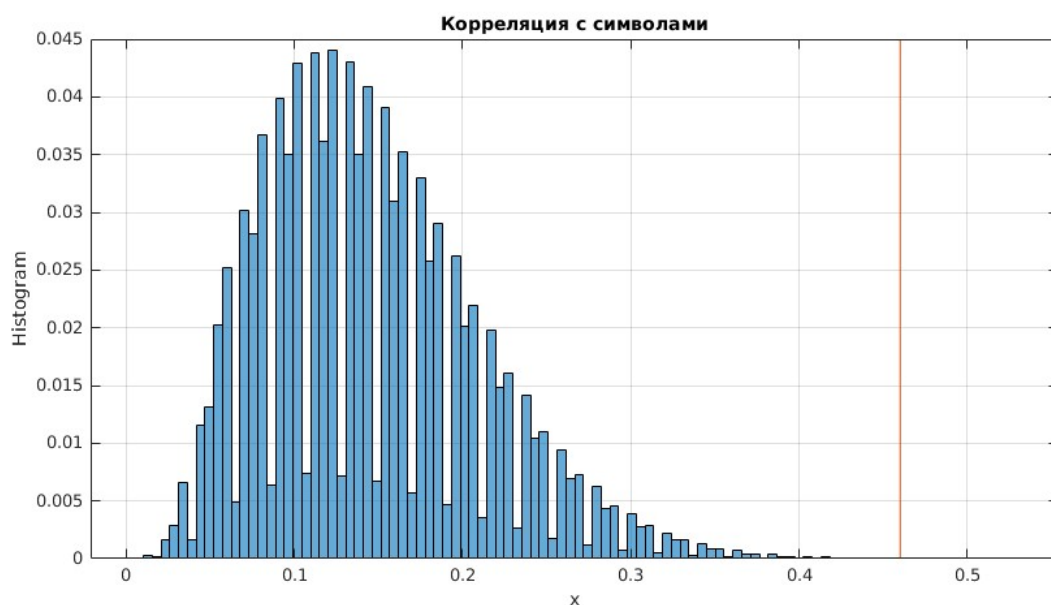


Рисунок 4.12 – Гистограмма максимумов корреляции с шумом по демодулированным символам.

Значения порога, обеспечивающее заданную вероятность ложной тревоги представлены в таблице 4.1. Для каждого из алгоритмов была построена экспериментальная характеристика обнаружения (рис. 4.13).

Таблица 4.1 – Значения порога для различных методов обнаружения.

Метод обнаружения	По комплексным отсчетам	По демодулированным отсчетам частоты	По демодулированным символам
Порог	0.33	0.34	0.46

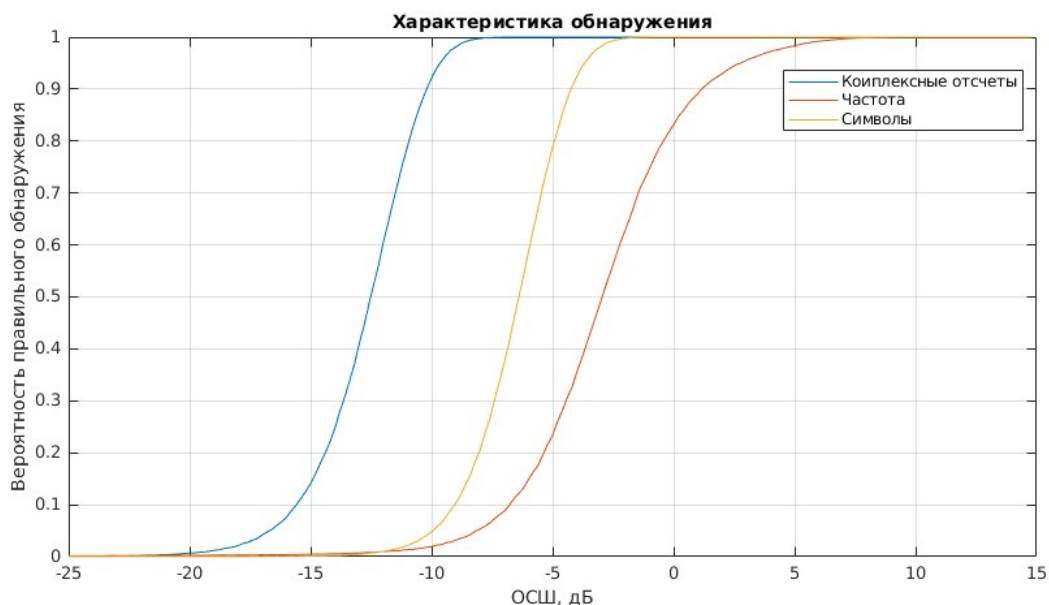


Рисунок 4.13 – Характеристики обнаружения для различных методов.

На основе полученных результатов можно сделать вывод о том, что обнаружение по комплексным отсчетам является наиболее приемлемым с точки зрения обнаружения. Для сравнения в таблице 4.2 приведены значения ОСШ для каждого из алгоритмов, при которых вероятность правильного обнаружения была равна 0.99.

Таблица 4.2 – Значения ОСШ для различных методов обнаружения.

Метод обнаружения	По комплексным отсчетам	По демодулированным отсчетам частоты	По демодулированным символам
ОСШ, дБ	-8.6	-2.7	5.8

## 4.5. Декодирование

Алгоритм декодирования реализован следующим образом:

- Первым этапом алгоритма декодирования является преобразование демодулированных символов в биты.
- Вторым этапом происходит декодирование поля «Тип слота». Для декодирования поля «Тип слота» используется декодер кода Рида-Соломона, реализация данного декодера является коммерческой тайной ООО «СТЦ». Из поля тип слота извлекаются информационные элементы «Цветовой код» и «Тип данных».
- Третьим этапом осуществляется деперемежение информационных бит. Класс деперемежитель *DmrDeinterleaver* написан мной на основании алгоритма перемежения описанного в стандарте DMR.
- Четвертым этапом является декодирование кода ВРТС (196, 96). Декодирование осуществляется с использованием декодера кода Хэмминга, реализация данного декодера является коммерческой тайной ООО «СТЦ».
- На пятом этапе происходит проверка CRC. Перед проверкой CRC к последним 24 битам информационного сообщения (битам четности) применяется специальная CRC маска, она выбирается на основе информационного элемента «Тип данных». В случае, если элемент «Тип данных» принимает значение Idle (Пустой), происходит формирование результата декодирования.
- Шестым этапом является получение системной информации. В зависимости от элемента «Тип данных» применяются разные варианты преобразования декодированных бит в системную информацию. Для преобразования системной информации используется класс *DmrParcer* написанный мной.

Описанный алгоритм реализован как метод *decodeData()* класса *DmrDeviceInfoExtractor* реализованного мной. Класс *DmrDeviceInfoExtractor* содержит в себе четыре объекта декодеров (декодер кода Хэмминга (13, 9), декодер кода Хэмминга (15, 11), декодер кода Голея (20,8) и декодер кода Рида-Соломона (12,9)), один объект класса *DmrDeinterleaver* и один объект класса *DmrParcer*. Реализация класса *DmrDeviceInfoExtractor* является коммерческой тайной ООО «СТЦ».

Блок-схема алгоритма декодирования представлена на рисунке 4.14.

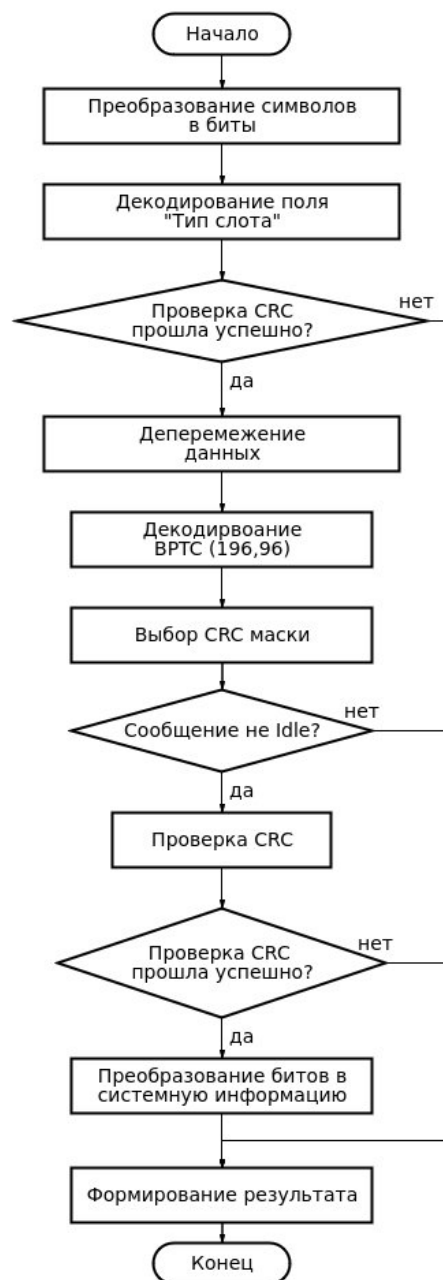


Рисунок 4.14 – Блок-схема алгоритма декодирования.

Верификация алгоритма декодирования проводилась путем сравнения бит, полученных после декодирования с соответствующими битами тестового сигнала. Аналогичные тестовые сигналы использовались при верификации кода на C++, в результате были получены аналогичные выходные данные.

#### 4.6. Быстродействие алгоритма

Для оценки быстродействия быстродействия алгоритма на вход программного модуля был подан тестовый сигнал размером 5120 отсчетов с частотой дискретизации 19200 Гц, что составляет 267 мс. Требования к быстродействию алгоритма выражены следующим условием: отношение времени обработки подаваемого сигнала ко времени самого сигнала не должно превышать 1:1.

Длительность обработки и отдельных её этапов представлены в таблице 4.3.

Таблица 4.3 – Быстродействие алгоритма.

Этап	Время, мкс
Коррекция частотной ошибки	19
Обнаружение	27
Демодуляция	21
Декодирование	5
Полный цикл	1725

Полученные результаты удовлетворяют заданным требованиям к быстродействию алгоритма, т.к. отношение времени обработки подаваемого сигнала ко времени самого сигнала равно 1:153.

#### 4.7. Помехоустойчивость алгоритма

Оценка помехоустойчивости алгоритма производилась с помощью имеющегося тестового сигнала, к которому был добавлен аддитивный белый

гауссовский шум. После обработки сигнала алгоритмом были собраны данные об обнаружении и декодировании сигнала. Для каждого значения ОСШ было произведено  $10^5$  итераций алгоритма, далее для оценки вероятности обнаружения сигнала при заданном ОСШ, суммарное количество обнаруженных сигналов, было разделено на количество итераций. График зависимости соответствующих оценочных вероятностей от ОСШ приведен на рис. 4.15.

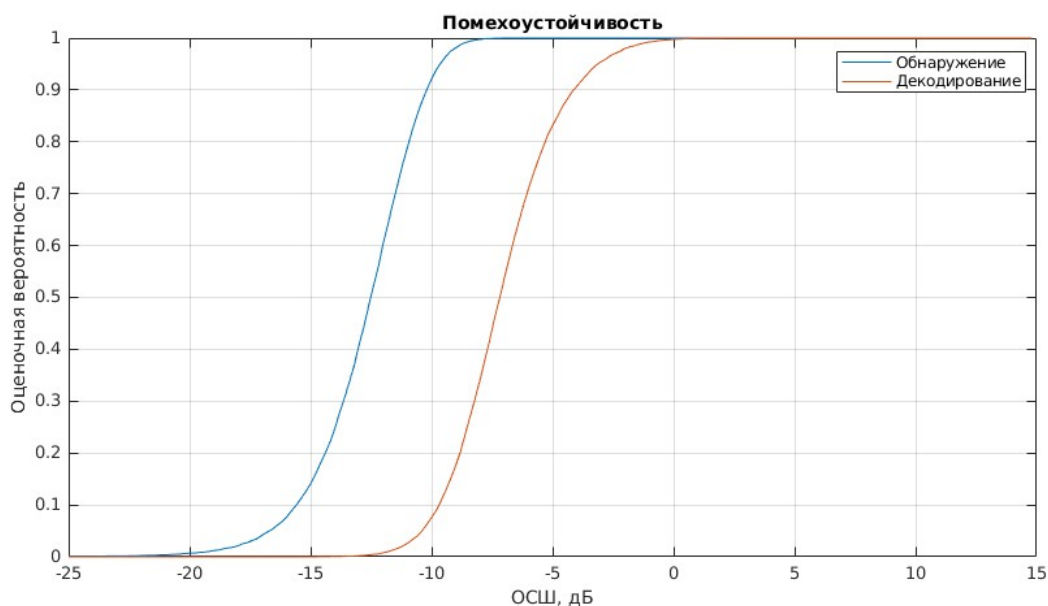


Рисунок 4.15 – Зависимость оценочных вероятностей от значения ОСШ

Данный график показывает, что реализованный алгоритм способен правильно декодировать исходные сообщения с вероятностью 0.9 при ОСШ не менее -4 дБ.

## **5. СПЕЦИАЛЬНЫЕ ВОПРОСЫ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ**

### **5.1. Используемое в разработке программное обеспечение**

В процессе разработки программного модуля было использовано следующее ПО:

- CLion – интегрированная среда разработки для языков программирования C и C++, разрабатываемая компанией JetBrains;
- MATLAB – пакет прикладных программ для решения задач технических вычислений.

Одной из существенных проблем безопасности в ПО является взаимосвязь между двумя объектами интерфейс/пользователь. Снижение риска воздействия вредных психофизиологических факторов достигается соблюдением эргономических требований. Основная цель эргономического проектирования ПО – создание хороших эргономических характеристик, которые позволяют исключить дискомфорт, обеспечивают безопасность и повышение производительности для обслуживающего персонала.

Одним из фундаментальных вопросов безопасности ПО является взаимодействие между двумя объектами интерфейс/пользователь. Эргономика необходима для снижения риска воздействия вредных психофизических факторов. Основная цель эргономичного проектирования ПО – создание хороших эргономических характеристик, которые устраняют дискомфорт, обеспечивают безопасность и повышают производительность обслуживающего персонала.

ПО используемое для разработки, отвечает следующим требованиям:

- пользовательский интерфейс является графическим;
- элементы управления визуализированы;
- пользователю предоставлена вся необходимая информация для принятия решения;

- ненужная для принятия решения информация исключена либо скрыта;
- пользователю предоставлена возможность легко различать наиболее вероятный вариант ответа;
- настройки программы являются понятными и простыми;
- дополнительные настройки скрыты от прямого доступа к ним с целью сохранения корректной работы программы.

Так же ПО удовлетворяет основным принципам организации диалога, приведенным в стандарте ИСО 9241-110, а именно:

- пригодность для выполнения задачи;
- информативность;
- соответствие ожиданиям пользователя;
- пригодность для обучения;
- управляемость;
- устойчивость к ошибкам;
- пригодность для индивидуализации.

ПО обеспечивает высокие показатели характеристик предоставления информации:

- информационное содержание передается быстро и точно;
- отображаемая информация может быть точно распознана;
- пользователям предоставляют только ту информацию, которая необходима для выполнения задачи;
- одинаковая информация представлена одинаковым образом во всем приложении, согласно ожиданиям пользователей;
- внимание пользователя направлено на требуемую информацию;
- информацию легко прочесть;
- значение информации понятно, недвусмысленно, интерпретируемо и узнаваемо.



Интерфейс CLion представлен на рис. 5.1. Основными элементами интерфейса являются: дерево проекта (слева), контроль версий (снизу) и основное окно, отображающее код программы.

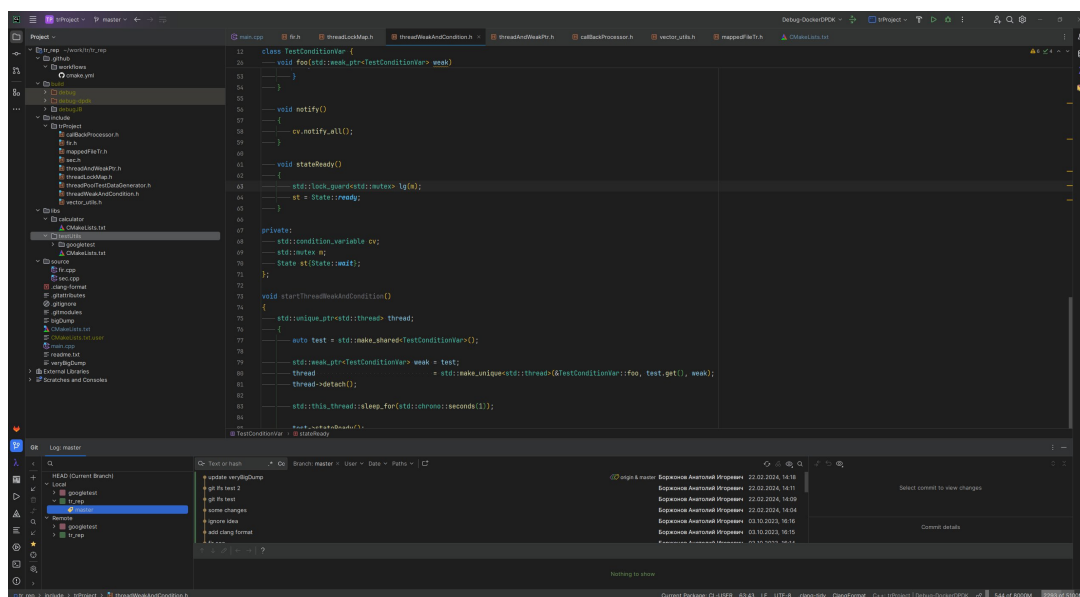


Рисунок 5.1 – Интерфейс ПО CLion.

ПО CLion обладает большой гибкостью настройки интерфейса. Так например дерево проекта может быть вынесено в отдельное окно, как и любой другой раздел. Помимо раздела контроля версий в CLion представлен большой набор инструментов, помогающих в разработке приложений на C и C++.

Интерфейс MATLAB представлен на рис. 5.2. Основными элементами интерфейса являются: меню (сверху), файловая система (слева), переменные и их значения (справа), вывод программы (снизу) и основное окно, отображающее код программы.



- поддержкой систем контроля версий Subversion, Mercurial, Git, CVS, Bazaar, Perforce;
- помощь при написании кода;
- возможностью разработки кроссплатформенных приложений.

Удобство данного ПО заключается в широком спектре возможностей по редактированию кода. В CLion реализовано автодополнение, в том числе ключевых слов, введённых в новейших стандартах C++, подсветка кода.

Реализован ряд возможностей при работе с сигнатурами методов, а именно:

- автогенерация пустого тела метода после его обновления;
- возможность автоматически изменить сигнатуру метода в определении, если она была изменена в объявлении и наоборот;
- возможность автоматически поменять порядок следования аргументов.

При навигации по коду доступно переключение между определением и объявлением метода, переход к объявлению метода, переименование метода, как в отдельном проекте, так и во всех открытых. Также есть возможность вызвать справку согласно текущему контексту.

Редактор с хорошим пониманием языка C и C++ позволяет работать с кодом более эффективно. Smart Completion позволяет фильтровать варианты автозаполнения по типу. Навигация breadcrumbs помогает понять, где вы находитесь в данный момент в иерархии пространства имен. Вызовы функций отображают всплывающие подсказки с информацией о параметрах. Также можно искать использование символов в контексте и переходить от имен символов к их определениям. CLion помогает проверить, следует ли ваш код определенному стилю, например, форматированию и соглашениям об именовании. Кроме того, можно переименовывать символы, встраивать функции, переменные и макросы, перемещать члены класса вверх по иерархии, изменять сигнатуры функций и извлекать переменные, параметры, функции и

объявления типов. CLion может выполнять анализ кода, анализ потока данных и другие проверки Clangd и Clang-Tidy. Удобный интерфейс отладчика CLion с поддержкой GDB и LLDB позволяет легко находить и решать проблемы в коде. CLion помогает легко находить и решать проблемы в коде. Имеется возможность подключаться к локальным процессам или отлаживать удаленно, а конфигурации OpenOCD и Embedded GDB Server помогают программировать микроконтроллеры и проверять периферийные устройства.

Пригодность ПО MATLAB к разработке программного модуля обеспечивается широким набором доступных инструментов по работе с числовыми и графическими данными.

MATLAB предоставляет удобные средства для разработки алгоритмов, включая высокоуровневые с использованием концепций объектно-ориентированного программирования. В нём имеются все необходимые средства интегрированной среды разработки, включая отладчик и профайлер. Функции для работы с целыми типами данных облегчают создание алгоритмов для микроконтроллеров и других приложений, где это необходимо.

В составе пакета MATLAB имеется большое количество функций для построения графиков, в том числе трёхмерных, визуального анализа данных и создания анимированных роликов. Встроенная среда разработки позволяет создавать графические интерфейсы пользователя с различными элементами управления, такими как кнопки, поля ввода и другими.

### **5.3. Выводы по 5 главе**

В данной главе были рассмотрены основные вопросы, касающиеся удобства и пригодности использования выбранного ПО для разработки программного модуля. Так же было показано, что используемое ПО соответствует основным требованиям ГОСТ Р ИСО 9241.

Поскольку результатом разработки программного модуля является серверное ПО, не имеющее интерфейса для взаимодействия с пользователем,

для его максимально эргономичного использования, время работы алгоритма было минимизировано.

## ЗАКЛЮЧЕНИЕ

В результате выполнения ВКР был реализован программный модуль обработки сигнала стандарта DMR. Программный модуль реализован на языке программирования C++ с использованием объектно-ориентированного стиля программирования и представляет собой высокопроизводительное, кроссплатформенное ПО, реализующее обнаружение, демодуляцию и декодирование системной информации о БС.

В ходе выполнения ВКР были решены следующие задачи:

1. Обзор литературы по вопросам цифровой связи и радиомониторинга;
2. Изучение стандарта DMR Air Interface protocol;
3. Изучение основных этапов формирования сигнала физического уровня;
4. Разработка алгоритма приема сигнала;
5. Реализация алгоритма на языке программирования C++;
6. Верификация и подбор параметров алгоритма;
7. Тестирование ПО на устройстве;

Так же были соблюдены все требования представленные к программному модулю: реализация с использованием объектно-ориентированного стиля программирования, вероятность ложной тревоги при обнаружении сигнала не более  $10^{-4}$ , быстродействие алгоритма не хуже чем 1:1 (1:153).

При реализации алгоритма обнаружения был использован критерий Неймана-Пирсона, в дальнейшем для усовершенствования работы алгоритма будут проведены исследования статистических характеристик реальных записей сигналов, а также проведена оценка возможных потерь при ложном обнаружении сигнала.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Сергиенко А.Б. Цифровая связь: учеб. пособие. СПб.: Изд-во СПб-ГЭТУ «ЛЭТИ», 2012. 164 с.
2. Склад Б. Цифровая связь. Теоретические основы и практическое применение. Изд. 2-е / под ред. А.В. Назаренко М.: Издательский дом "Вильямс", 2003. 1104 с.
3. Прокис Дж. Цифровая связь / под ред. Д.Д. Кловского М.: Радио и связь, 2000. 800 с.
4. Радиомониторинг – задачи, методы, средства / Под ред. А.М. Рембовского. 2-е изд., перераб. И доп. – М.: Горячая линия-Телеком, 2021. – 624 с.: ил.
5. Справочник по радиоконтролю. МСЭ 2002. Женева. 2004. 584 с.
6. А.А. Привалов, П.Н. Ерлыков Методы защиты сетей подвижной радиосвязи стандарта DMR от случайных и преднамеренных помех. / 75-я научно-техническая конференция Санкт-Петербургского НТО РЭС им. А.С. Попова, посвященная Дню радио, СПб 20-24 апреля 2020 г. / СПбГЭТУ «ЛЭТИ», СПб, 2020, С. 191–193.
7. ETSI TS 102 361-1 V2.5.1. Electromagnetic compatibility and Radio spectrum Matters (ERM); Digital Mobile Radio (DMR) Systems; Part 1: DMR Air Interface protocol. ETSI, 2017.
8. ETSI TS 102 361-1 V2.3.1. Electromagnetic compatibility and Radio spectrum Matters (ERM); Digital Mobile Radio (DMR) Systems; Part 2: DMR voice and generic services and facilities. ETSI, 2016.
9. Сергиенко А.Б. Цифровая обработка сигналов – СПб .: Питер, 2002. - 608 с.: с ил.